US011875028B2

(12) **United States Patent**
Pedrick et al.

(10) **Patent No.: US 11,875,028 B2**
(45) **Date of Patent: *Jan. 16, 2024**

(54) **NATIVE APPLICATION COLLABORATION**

(71) Applicant: **Dropbox, Inc.**, San Francisco, CA (US)

(72) Inventors: **Martin Pedrick**, New York, NY (US); **Michael Wu**, San Francisco, CA (US); **Anvisha Pai**, San Francisco, CA (US)

(73) Assignee: **DROPBOX, INC.**, San Francisco, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 408 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **16/811,893**

(22) Filed: **Mar. 6, 2020**

(65) **Prior Publication Data**

US 2020/0210058 A1 Jul. 2, 2020

Related U.S. Application Data

(63) Continuation of application No. 14/983,847, filed on Dec. 30, 2015, now Pat. No. 10,620,811.

(51) **Int. Cl.**

G06F 3/04847	(2022.01)
G06F 16/178	(2019.01)
H04L 67/50	(2022.01)
G06F 3/0481	(2022.01)

(52) **U.S. Cl.**

CPC ***G06F 3/04847*** (2013.01); ***G06F 3/0481*** (2013.01); ***G06F 16/178*** (2019.01); ***H04L 67/535*** (2022.05)

(58) **Field of Classification Search**

CPC .. G06F 3/04847; G06F 3/0481; G06F 16/178; G06F 16/1734; H04L 67/535

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,337,407	A	8/1994	Bates et al.
5,805,889	A	9/1998	Van De Vanter
5,960,173	A	9/1999	Tang et al.
6,065,026	A	5/2000	Cornelia et al.
6,088,707	A	7/2000	Bates et al.

(Continued)

FOREIGN PATENT DOCUMENTS

CN	102342080	A	2/2012
JP	H06214743	A	8/1994

(Continued)

OTHER PUBLICATIONS

Non-Final Office Action from U.S. Appl. No. 16/510,018, dated Mar. 19, 2021, 14 pages.

(Continued)

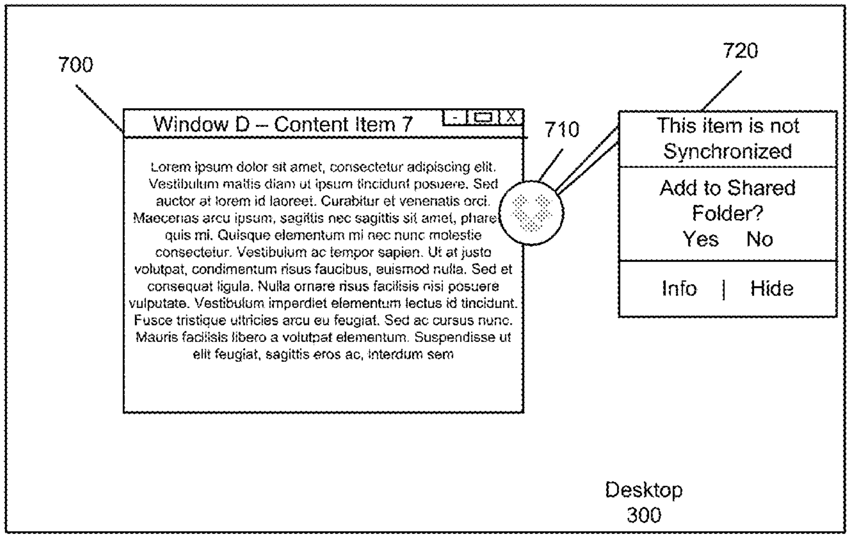
Primary Examiner — Jennifer E Nichols

(74) *Attorney, Agent, or Firm* — Fenwick & West LLP

(57) **ABSTRACT**

A client device displays a content item in a window associated with a process on the device. A client application monitors the process for the content item and determines whether the content item is synchronized with a client device or content management system. When the content item is not synchronized, a synchronization interface is displayed to permit a user to request synchronizing the content item. When the user requests synchronization, the content item is moved to a shared folder synchronized with the content management system or client device.

17 Claims, 13 Drawing Sheets



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, phare quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

(56) References Cited

U.S. PATENT DOCUMENTS

6,260,040	B1	7/2001	Kauffman et al.	
6,910,188	B2 *	6/2005	Keohane	G06Q 10/10 715/741
6,938,042	B2	8/2005	Aboulhosn et al.	
7,062,515	B1 *	6/2006	Thomas	H04L 67/1095 707/951
7,089,307	B2	8/2006	Zintel et al.	
7,386,529	B2	6/2008	Kiessig et al.	
7,401,104	B2	7/2008	Shah et al.	
7,496,633	B2	2/2009	Szeto et al.	
7,533,134	B2	5/2009	Terry et al.	
7,558,797	B2 *	7/2009	Li	G06F 16/14
7,603,357	B1 *	10/2009	Gourdol	G06F 16/1774 718/107
7,610,280	B2	10/2009	O'Toole et al.	
7,620,659	B2	11/2009	Novik et al.	
7,631,007	B2	12/2009	Morris et al.	
7,640,506	B2	12/2009	Pratley et al.	
7,676,526	B1	3/2010	Beloussov et al.	
7,769,810	B1	8/2010	Kaufman	
7,797,637	B2 *	9/2010	Marcjan	G06F 3/0486 715/764
7,818,419	B1	10/2010	McAllister et al.	
7,822,711	B1	10/2010	Ranade et al.	
7,912,916	B2	3/2011	Rakowski et al.	
8,019,877	B2	9/2011	Li et al.	
8,041,807	B2	10/2011	Bao et al.	
8,055,644	B2 *	11/2011	Crowley	G06F 16/182 707/709
8,086,698	B2	12/2011	Rakowski et al.	
8,140,513	B2	3/2012	Ghods et al.	
8,156,078	B2	4/2012	Tsinman et al.	
8,205,150	B2 *	6/2012	Gelman	G06F 40/197 715/229
8,230,348	B2	7/2012	Peters et al.	
8,244,830	B2	8/2012	Robinson et al.	
8,301,606	B2	10/2012	Futatsugi et al.	
8,386,558	B2	2/2013	Schleifer et al.	
8,452,959	B2	5/2013	Brown et al.	
8,468,132	B1	6/2013	O'Neill et al.	
8,495,015	B2	7/2013	Freedman et al.	
8,521,732	B2	8/2013	Wood et al.	
8,543,637	B2	9/2013	Kent et al.	
8,555,177	B1	10/2013	Junee et al.	
8,600,934	B2	12/2013	Strong et al.	
8,615,552	B2	12/2013	Shah et al.	
8,635,209	B2	1/2014	Freedman et al.	
8,650,498	B1 *	2/2014	Mihovilovic	G06F 16/178 709/248
8,655,840	B2	2/2014	Simelius et al.	
8,700,759	B2	4/2014	Chakra et al.	
8,719,225	B1	5/2014	Rath et al.	
8,825,597	B1 *	9/2014	Houston	G06F 16/10 707/625
8,843,441	B1	9/2014	Rath et al.	
8,868,500	B2	10/2014	Giampaolo et al.	
8,930,312	B1	1/2015	Rath et al.	
8,930,488	B2	1/2015	Dantec et al.	
8,965,847	B1 *	2/2015	Chang	G06F 16/192 707/615
8,997,007	B1	3/2015	Bennett et al.	
9,021,113	B2	4/2015	Carr et al.	
9,053,165	B2	6/2015	Van Rossum	
9,240,962	B2	1/2016	Jung et al.	
9,280,613	B2 *	3/2016	Smith	G06F 16/14
9,298,355	B1	3/2016	Beausoleil et al.	
9,298,834	B2	3/2016	Kleppner et al.	
9,307,006	B2	4/2016	Micucci et al.	
9,338,242	B1	5/2016	Suchland et al.	
9,413,708	B1 *	8/2016	Michael	H04L 51/224
9,432,457	B2 *	8/2016	Marano	G06F 16/178
9,519,886	B2 *	12/2016	Berger	H04L 65/403
9,575,981	B2 *	2/2017	Dorman	H04L 67/06
9,602,560	B1	3/2017	Moody et al.	
9,614,880	B1	4/2017	Davis et al.	
9,646,023	B1 *	5/2017	McCabe	G06F 11/14
9,652,741	B2 *	5/2017	Goldberg	G06F 16/178
9,690,785	B1	6/2017	Vagell	
9,773,010	B1	9/2017	Ganesh et al.	
9,773,051	B2	9/2017	Smith	
9,832,237	B2 *	11/2017	Lock	H04L 65/403
9,998,555	B2	6/2018	Larabie-Belanger	
10,091,287	B2	10/2018	Larabie-Belanger	
10,110,656	B2 *	10/2018	Sawyer	G06F 16/1774
10,133,720	B2	11/2018	Rothschiller et al.	
10,171,579	B2	1/2019	Larabie-Belanger	
10,235,022	B2	3/2019	Eccleston et al.	
10,270,871	B2	4/2019	Larabie-Belanger	
10,318,492	B2	6/2019	Eisner	
10,620,811	B2 *	4/2020	Pedrick	H04L 67/535
11,244,284	B2 *	2/2022	Milvaney	G06F 16/93
11,347,762	B2	5/2022	Beausoleil et al.	
2001/0042073	A1	11/2001	Saether et al.	
2002/0049786	A1	4/2002	Bibliowicz et al.	
2003/0018719	A1	1/2003	Ruths et al.	
2003/0112273	A1	6/2003	Hadfield et al.	
2003/0159136	A1	8/2003	Huang et al.	
2004/0021686	A1	2/2004	Barberis	
2004/0024795	A1	2/2004	Hind et al.	
2004/0058710	A1	3/2004	Timmins et al.	
2004/0172425	A1 *	9/2004	Edelstein	G06F 40/194 707/999.203
2004/0199514	A1	10/2004	Rosenblatt et al.	
2004/0221010	A1	11/2004	Butler	
2005/0022210	A1	1/2005	Zintel et al.	
2005/0044187	A1	2/2005	Jhaveri et al.	
2005/0050067	A1 *	3/2005	Sollicito	H04L 67/142
2005/0192822	A1	9/2005	Hartenstein et al.	
2005/0223047	A1	10/2005	Shah et al.	
2005/0246544	A1	11/2005	Moore et al.	
2005/0278392	A1	12/2005	Hansen et al.	
2006/0004911	A1	1/2006	Becker et al.	
2006/0015863	A1 *	1/2006	Vaidyanathan	G06F 9/451 707/999.203
2006/0101064	A1	5/2006	Strong et al.	
2006/0155776	A1 *	7/2006	Aust	G06F 16/10
2006/0190117	A1	8/2006	Weczorek et al.	
2006/0235984	A1	10/2006	Kraus et al.	
2006/0259957	A1	11/2006	Tam et al.	
2006/0288053	A1	12/2006	Holt et al.	
2007/0016650	A1 *	1/2007	Gilbert	G09B 5/00 709/207
2007/0027853	A1	2/2007	Hajela et al.	
2007/0055926	A1 *	3/2007	Christiansen	G06Q 10/10 715/244
2007/0143324	A1 *	6/2007	Eichhorst	G06F 16/168
2007/0150433	A1 *	6/2007	Chen	G06F 8/71
2007/0168490	A1	7/2007	Kent et al.	
2007/0174787	A1	7/2007	Rhee et al.	
2007/0271502	A1	11/2007	Bedi et al.	
2008/0005188	A1	1/2008	Li et al.	
2008/0005235	A1	1/2008	Hegde et al.	
2008/0109547	A1	5/2008	Bao et al.	
2008/0178117	A1 *	7/2008	Gelman	G06F 3/0481 707/999.203
2008/0195759	A1	8/2008	Novik et al.	
2008/0222296	A1	9/2008	Lippincott et al.	
2008/0243807	A1	10/2008	Gaucas et al.	
2008/0244418	A1	10/2008	Manolescu et al.	
2008/0307102	A1	12/2008	Galloway et al.	
2009/0006498	A1	1/2009	Freedman et al.	
2009/0113412	A1	4/2009	Shribman et al.	
2009/0138808	A1	5/2009	Moromisato et al.	
2009/0157627	A1	6/2009	Arthursson	
2009/0222763	A1	9/2009	Dukhon et al.	
2009/0228504	A1	9/2009	Brown et al.	
2009/0249224	A1	10/2009	Davis et al.	
2009/0271696	A1	10/2009	Bailor et al.	
2009/0271708	A1	10/2009	Peters et al.	
2009/0288150	A1	11/2009	Toomim et al.	
2009/0292681	A1	11/2009	Wood et al.	
2009/0292709	A1	11/2009	Tsinman et al.	
2009/0313311	A1	12/2009	Hoffmann et al.	

(56) References Cited

U.S. PATENT DOCUMENTS

2009/0327358	A1	12/2009	Lukiyanov et al.	
2010/0017879	A1	1/2010	Kuegler et al.	
2010/0083136	A1	4/2010	Komine et al.	
2010/0131868	A1	5/2010	Chawla et al.	
2010/0138387	A1	6/2010	Simelius et al.	
2010/0191884	A1	7/2010	Holenstein et al.	
2010/0198784	A1	8/2010	Serlet	
2010/0198874	A1	8/2010	Futatsugi et al.	
2010/0251340	A1	9/2010	Martin et al.	
2010/0257450	A1	10/2010	Go et al.	
2010/0257457	A1	10/2010	De Goes	
2010/0281384	A1	11/2010	Lyons et al.	
2011/0016450	A1*	1/2011	Karadakal	G06F 8/71 717/116
2011/0055177	A1	3/2011	Chakra et al.	
2011/0106880	A1	5/2011	Strong et al.	
2011/0113092	A1	5/2011	Rakowski et al.	
2011/0208805	A1	8/2011	Kasetty et al.	
2011/0231499	A1	9/2011	Stovicek et al.	
2011/0296042	A1	12/2011	Li et al.	
2012/0005159	A1	1/2012	Wang et al.	
2012/0054279	A1	3/2012	Dantec et al.	
2012/0089710	A1	4/2012	Rakowski et al.	
2012/0101980	A1	4/2012	Taleghani et al.	
2012/0124486	A1	5/2012	Robinson et al.	
2012/0130953	A1	5/2012	Hind et al.	
2012/0151495	A1	6/2012	Burckhardt et al.	
2012/0185434	A1	7/2012	Giampaolo et al.	
2012/0227059	A1*	9/2012	Fleming	G06T 11/20 719/318
2012/0233554	A1	9/2012	Vagell et al.	
2012/0265836	A1*	10/2012	Nemoto	G06F 16/487 709/211
2012/0278127	A1	11/2012	Kirakosyan et al.	
2012/0303716	A1	11/2012	Peters et al.	
2012/0317239	A1	12/2012	Mulder et al.	
2012/0324121	A1	12/2012	Carr et al.	
2012/0331108	A1	12/2012	Ferdowsi et al.	
2013/0006947	A1	1/2013	Akinyemi et al.	
2013/0013560	A1*	1/2013	Goldberg	G06F 16/178 707/634
2013/0054509	A1	2/2013	Kass et al.	
2013/0060859	A1	3/2013	Olsen et al.	
2013/0091205	A1*	4/2013	Kotler	H04L 67/54 709/204
2013/0117376	A1	5/2013	Filman et al.	
2013/0124638	A1*	5/2013	Barreto	G06F 16/27 709/205
2013/0124967	A1	5/2013	Hatfield et al.	
2013/0124978	A1	5/2013	Horns et al.	
2013/0138608	A1*	5/2013	Smith	G06F 16/27 707/610
2013/0151596	A1	6/2013	Strong et al.	
2013/0185651	A1	7/2013	Wionzek et al.	
2013/0191339	A1	7/2013	Haden et al.	
2013/0191451	A1*	7/2013	Tse	H04L 67/54 709/204
2013/0204841	A1	8/2013	Obusek et al.	
2013/0212250	A1*	8/2013	Kleppner	G06F 16/9535 709/224
2013/0212480	A1	8/2013	Ben-Shaul et al.	
2013/0238558	A1*	9/2013	Barefoot	G06F 16/178 707/638
2013/0247004	A1	9/2013	DeLuca et al.	
2013/0254656	A1	9/2013	Florian et al.	
2013/0268480	A1*	10/2013	Dorman	G06F 16/27 707/608
2013/0275509	A1	10/2013	Micucci et al.	
2013/0282656	A1	10/2013	O'Neill et al.	
2013/0297680	A1*	11/2013	Smith	H04L 67/535 709/217
2013/0332523	A1	12/2013	Luu et al.	
2014/0019626	A1	1/2014	Hubler et al.	
2014/0025712	A1*	1/2014	Chisa	G06F 9/4451 707/827
2014/0040202	A1*	2/2014	Hagan	G06F 16/27 707/634
2014/0059002	A1*	2/2014	Lockhart	G06F 16/178 707/622
2014/0068629	A1	3/2014	Boller et al.	
2014/0101310	A1*	4/2014	Savage	H04W 4/60 709/224
2014/0129645	A1	5/2014	Mo	
2014/0156749	A1	6/2014	Kanter et al.	
2014/0160153	A1	6/2014	Singh	
2014/0172799	A1*	6/2014	Dorman	G06F 16/178 707/638
2014/0201138	A1*	7/2014	Dorman	G06F 16/178 707/610
2014/0201144	A1*	7/2014	Vibhor	G06F 16/178 707/634
2014/0201289	A1	7/2014	Wheeler et al.	
2014/0215302	A1*	7/2014	Little	G06F 40/166 715/256
2014/0223333	A1	8/2014	Pegg et al.	
2014/0229543	A1	8/2014	Allison et al.	
2014/0229839	A1*	8/2014	Lynch	G06F 9/44505 715/736
2014/0237387	A1	8/2014	Ryall et al.	
2014/0244580	A1*	8/2014	Eisner	G06F 16/178 707/623
2014/0244583	A1	8/2014	Wang et al.	
2014/0244618	A1*	8/2014	Lynch	G06F 16/41 707/769
2014/0250064	A1*	9/2014	Hopmann	H04L 67/1095 707/623
2014/0279846	A1	9/2014	Srinivasan et al.	
2014/0280377	A1	9/2014	Frew	
2014/0280603	A1	9/2014	Rideout et al.	
2014/0289645	A1	9/2014	Megiddo et al.	
2014/0310345	A1*	10/2014	Megiddo	G06Q 10/101 709/204
2014/0330776	A1*	11/2014	Chen	H04L 67/02 707/610
2014/0372852	A1	12/2014	Rothschiller et al.	
2014/0379586	A1*	12/2014	Sawyer	G06F 16/168 705/301
2014/0379647	A1*	12/2014	Smith	G06F 16/178 707/624
2015/0012488	A1	1/2015	Van Rossum	
2015/0012887	A1	1/2015	Ash et al.	
2015/0032686	A1	1/2015	Kuchoor	
2015/0040101	A1	2/2015	Rummler et al.	
2015/0077326	A1	3/2015	Kramer et al.	
2015/0082196	A1	3/2015	Berger et al.	
2015/0106750	A1	4/2015	Konami et al.	
2015/0112927	A1*	4/2015	Lee	G06F 16/178 707/610
2015/0142742	A1	5/2015	Hong	
2015/0149404	A1	5/2015	Lock et al.	
2015/0261411	A1*	9/2015	Snook	G06F 3/04817 715/751
2015/0278330	A1*	10/2015	Hawa	G06F 16/275 709/203
2015/0281869	A1	10/2015	Ramachandran et al.	
2015/0286533	A1*	10/2015	Kaufthal	G06F 3/04817 707/639
2015/0286655	A1	10/2015	Von et al.	
2015/0288757	A1	10/2015	Larabie-Belanger	
2015/0288774	A1	10/2015	Larabie-Belanger et al.	
2015/0288775	A1	10/2015	Larabie-Belanger	
2015/0356111	A1*	12/2015	Kalsi	G06F 16/178 707/610
2015/0370769	A1	12/2015	Pereira et al.	
2015/0373116	A1*	12/2015	Mo	H04L 67/51 709/219
2015/0379879	A1	12/2015	Selkirk et al.	
2016/0078237	A1	3/2016	Tan	
2016/0105859	A1*	4/2016	Zhu	H04W 56/003 370/350

(56) **References Cited**

U.S. PATENT DOCUMENTS

2016/0119414	A1	4/2016	Westaway et al.	
2016/0162370	A1*	6/2016	Mehta	G06F 16/162 707/610
2016/0253352	A1*	9/2016	Kluck	H04L 67/1095 707/638
2016/0259508	A1	9/2016	Eccleston et al.	
2016/0308940	A1	10/2016	Procopio et al.	
2016/0321227	A1*	11/2016	Keslin	G06F 3/04842
2016/0350327	A1*	12/2016	Fan	G06F 16/16
2017/0093777	A1*	3/2017	Neustifter	H04L 67/535
2017/0192656	A1*	7/2017	Pedrick	H04L 67/535
2017/0249328	A1	8/2017	Liang et al.	
2018/0062852	A1	3/2018	Schmahmann	
2018/0113862	A1*	4/2018	Glover	G06F 16/1873
2018/0234496	A1*	8/2018	Ratias	H04L 67/535

FOREIGN PATENT DOCUMENTS

JP	H10187619	A	7/1998
JP	2002351730	A	12/2002
JP	2003157225	A	5/2003
JP	2003178015	A	6/2003
JP	2007500407	A	1/2007
JP	2008210192	A	9/2008
JP	2010517122	A	5/2010
JP	2010211569	A	9/2010
JP	2010218432	A	9/2010
JP	2011526034	A	9/2011
JP	2012038210	A	2/2012
JP	2012150647	A	8/2012
JP	2012519890	A	8/2012
JP	2012173780	A	9/2012
JP	2012209614	A	10/2012
JP	2013161481	A	8/2013
JP	2013196233	A	9/2013
JP	2014095960	A	5/2014
JP	2015505627	A	2/2015
JP	2015513713	A	5/2015
KR	20110124797	A	11/2011
KR	20150069397	A	6/2015
WO	2005114383	A2	12/2005
WO	2010100354	A1	9/2010
WO	2011109416	A2	9/2011
WO	2013109552	A1	7/2013
WO	2013109557	A1	7/2013
WO	2014003781	A1	1/2014
WO	2014021031	A1	2/2014
WO	2015139179	A1	9/2015

OTHER PUBLICATIONS

Notice of Allowance from U.S. Appl. No. 16/560,901, dated Nov. 27, 2020, 8 pages.

Anonymous: "Event (computing)—Wikipedia," XP055728697, Apr. 3, 2014, retrieved from the Internet: URL:https://en.wikipedia.org/w/index.php?title=Event_(computing)&oldid=602539019 on Sep. 8, 2020, 5 pages.

Bishop J., et al., "Developing Principles of GUI Programming using Views," Computer Science Education, ACM, Mar. 1, 2004, pp. 373-377.

Brief Communication of Oral Proceedings on Sep. 28, 2020 for European application No. 15736316.9 dated Sep. 16, 2020, 4 pages.

Bruce K.B., et al., "Event-Driven Programming Facilitates Learning Standard Programming Concepts," Object-Oriented Programming Systems, Languages and Applications, ACM, Oct. 23, 2004, pp. 96-100.

Examination Report No. 1, for Australian Application No. 2019284033, dated Dec. 17, 2020, 5 pages.

Ganov S., et al., "Event Listener Analysis and Symbolic Execution for Testing GUI Applications," Formal Methods and Software Engineering, Dec. 9, 2009, pp. 69-87.

Adahl, J., "Shared Resource for Collaborative Editing Over a Wireless Network," Master of Science Thesis, University of Gothenburg, Dec. 2010, 274 pages.

Anonymous, "c#—Is there Windows System Event on Active Window Changed?" Stack Overflow, Nov. 15, 2015, retrieved from https://web.archive.org/web/20151115134250/https://stackoverflow.com/questions/4407631/is-there-windows-system-eventon-active-wi ndow-changed on Mar. 25, 2019, 1 page.

Decision of Refusal for Japanese Application No. 2018-517385 dated Nov. 11, 2019, 7 pages.

European Summons to Attend Oral Proceedings Pursuant to Rule 115(1) EPC for EP Application No. 15736317.7 dated Oct. 24, 2019, 7 pages.

European Summons to Attend Oral Proceedings Pursuant to Rule 115(1) EPC for EP Application No. 16767369.8 dated Nov. 6, 2019, 8 pages.

European Summons to Attend Oral Proceedings Pursuant to Rule 115(1) EPC for EP Application No. 15736316.9 dated Oct. 24, 2019, 7 pages.

Examination Report No. 4, for Australian Application No. 2016381225, dated Nov. 28, 2019, 5 pages.

Examination Report for EP Application No. 15736316.9 dated Oct. 30, 2017, 5 pages.

Examination Report for EP Application No. 15736317.7 dated Oct. 30, 2017, 4 pages.

Examination Report No. 1, for Australian Application No. 2015246019, dated Apr. 11, 2019, 4 pages.

Examination Report No. 3, for Australian Application No. 2016381225, dated Sep. 2, 2019, 4 pages.

Examination Report No. 1 for Australian Application No. 2016381225 dated Feb. 28, 2019, 4 pages.

First Office Action for Japanese Application No. 2018517385 dated Mar. 18, 2019, 5 pages.

International Search Report and Written Opinion for PCT Application No. PCT/IB2015/052519 dated Jun. 18, 2015, 9 pages.

International Search Report and Written Opinion for PCT Application No. PCT/IB2015/052521 dated Sep. 24, 2015, 8 pages.

International Search Report and Written Opinion for PCT Application No. PCT/IB2015/052522 dated Jun. 18, 2015, 11 pages.

International Search Report and Written Opinion for PCT Application No. PCT/IB2015/052523 dated Sep. 24, 2015, 7 pages.

International Search Report and Written Opinion for PCT Application No. PCT/IB2016/055286 dated Oct. 31, 2016, 12 pages.

Lim H., et al., "Ubi-Jector", CHI 2013: Changing Perspectives, Paris, France, Apr. 27-May 2, 2013, pp. 1695-1700.

Non-Final Office Action from U.S. Appl. No. 14/983,847, dated Oct. 31, 2019, 14 pages.

Non-Final Office Action from U.S. Appl. No. 15/867,693, dated Sep. 17, 2019, 19 pages.

Non-Final Office Action from U.S. Appl. No. 16/115,563, dated Oct. 3, 2019, 8 pages.

Notice of acceptance for Australian Application No. 2015246018 dated Oct. 21, 2019, 3 pages.

Notice of Acceptance for Australian Application No. 2016381225, dated Feb. 13, 2020, 3 pages.

Notice of Allowance from U.S. Appl. No. 14/983,847, dated Jan. 30, 2020, 10 pages.

Notice of Allowance for Japanese Application No. 2018517385, dated Mar. 27, 2020, 5 pages.

Office Action for Australian Application No. 2015246018, dated Apr. 5, 2019, 6 pages.

Office Action for European Application No. 16767369.8 dated Apr. 1, 2019, 7 pages.

Office Action for Japanese Application No. 2016-553439 dated Oct. 19, 2018, 7 pages.

Office Action for Japanese Application No. 2016-553452 dated Oct. 19, 2018, 11 pages.

Office Action for Japanese Application No. 2016-553454 dated Nov. 16, 2018, 5 pages.

Office Action for U.S. Appl. No. 16/296,104 dated Sep. 6, 2019, 13 pages.

Second Office Action for Australia Application No. 2016381225, dated May 10, 2019, 6 pages.

(56) **References Cited**

OTHER PUBLICATIONS

Terry D.B., et al., "Managing Update Conflicts in Bayou, A Weakly Connected Replicated Storage System," Proceedings of the 15th ACM Symposium on Operating System Principles, Dec. 3-6, 1995, pp. 172-183.

"Document Object Model Events," retrieved from https:/ /web.archive.org/web/20120218183544/http://www.w3.org/TR/DOM-Levei-2-Events/events.html, Feb. 18, 2012, 23 pages.

Examination Report No. 1, for Australian Application No. 2019264563, dated Nov. 10, 2020, 6 pages.

"Listeners Supported by Swing Components," retrieved from https:/ /web.archive.org/web/20131010082743/http:/ /docs.oracle.com/javase/tutorial/uiswing/events/eventsandcomponents.html, Oct. 10, 2013, 3 pages.

Notice of Acceptance for Australian Application No. 2019257465, dated Jun. 22, 2021, 3 pages.

Notice of Allowance from U.S. Appl. No. 16/257,503, dated Jul. 1, 2021, 16 pages.

Examination Report No. 2, for Australian Application No. 2019264563, dated Feb. 22, 2021, 4 pages.

Examination Report No. 2, for Australian Application No. 2019284033, dated Feb. 10, 2021, 6 pages.

Final Office Action from U.S. Appl. No. 16/257,503, dated Mar. 12, 2021, 22 pages.

Non-Final Office Action from U.S. Appl. No. 16/296,104, dated Jul. 22, 2020, 13 pages.

Non-Final Office Action from U.S. Appl. No. 16/510,018, dated Jul. 23, 2020, 19 pages.

Non-Final Office Action from U.S. Appl. No. 16/784,747, dated Oct. 15, 2020, 19 pages.

Notice of Allowance for Japanese Application No. 2019-036310 dated Oct. 16, 2020, 5 pages.

Final Office Action from U.S. Appl. No. 16/296,104, dated Oct. 1, 2020, 12 pages.

Non-Final Office Action from U.S. Appl. No. 16/257,503, dated Oct. 1, 2020, 16 pages.

Notice of Allowance from U.S. Appl. No. 16/560,901, dated Sep. 14, 2020, 8 pages.

Notice of Allowance from U.S. Appl. No. 16/560,901, dated Sep. 24, 2020, 5 pages.

Notice of Allowance for Japanese Application No. 2019-124719, dated Oct. 5, 2020, 5 pages.

Notice of allowance of Japanese Application No. 2019-080438, dated Jun. 1, 2020, 5 pages.

Notice of allowance of Japanese Application No. 2019-029541, dated Jun. 8, 2020, 5 pages.

Final Office Action from U.S. Appl. No. 16/510,018, dated Sep. 23, 2021, 16 pages.

Notice of Allowance from U.S. Appl. No. 16/280,845, dated Sep. 14, 2021, 9 pages.

Notice of Allowance from U.S. Appl. No. 16/296,104, dated Sep. 23, 2021, 16 pages.

Non-Final Office Action from U.S. Appl. No. 17/501,680, dated Jun. 9, 2022, 17 pages.

Notice of Allowance from U.S. Appl. No. 16/510,018, dated Jun. 7, 2022, 7 pages.

Examination Report No. 2 for Australian Application No. 2019257465 dated Dec. 18, 2020, 4 pages.

Notice of Allowance from U.S. Appl. No. 16/784,747, dated Feb. 9, 2021, 8 pages.

Notice of Reasons for Refusal for Japanese Application No. 2019-036310 dated Mar. 16, 2020, 7 pages.

Notice of Allowance from U.S. Appl. No. 15/867,693, dated Aug. 13, 2020, 2 pages.

Notice of Allowance from U.S. Appl. No. 15/867,693, dated Aug. 21, 2020, 2 pages.

1 Non-Final Office Action from U.S. Appl. No. 16/510,018, dated Jan. 11, 2022, 15 pages.

Advisory Action from U.S. Appl. No. 16/257,503, dated Apr. 21, 2021, 3 pages.

Examination Report No. 3 for Australian Application No. 2019257465, dated Mar. 23, 2021, 6 pages.

Advisory Action from U.S. Appl. No. 16/296,104, dated Apr. 14, 2020, 3 pages.

Examination Report No. 2, for Australian Application No. 2015246019, dated Jul. 10, 2019, 3 pages.

Final Office Action from U.S. Appl. No. 15/867,693, dated Jan. 28, 2020, 17 pages.

Final Office Action from U.S. Appl. No. 16/257,503, dated Jun. 25, 2020, 17 pages.

Final Office Action from U.S. Appl. No. 16/296,104, dated Feb. 19, 2020, 13 pages.

First Office Action for Australian Application No. 2015246020, dated Apr. 12, 2019, 4 pages.

First Office Action for Japanese Application No. 2018517385, dated Mar. 18, 2019, 5 pages (with English concise explanation of relevance).

International Search Report and Written Opinion for PCT Application No. PCT/US2014/045631 dated Oct. 6, 2014, 14 pages.

Non-Final Office Action from U.S. Appl. No. 16/257,503, dated Feb. 14, 2020, 18 pages.

Notice of Allowance from U.S. Appl. No. 16/115,563, dated Dec. 13, 2019, 7 pages.

Notice of Reasons for Refusal for Japanese Application No. 2018-517385, dated Mar. 18, 2019, 7 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Apr. 4, 2018, 16 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Dec. 14, 2017, 19 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Dec. 30, 2016, 12 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Feb. 20, 2018, 4 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Jun. 17, 2016, 15 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Jun. 29, 2017, 17 pages.

Office Action for U.S. Appl. No. 14/248,158, dated Oct. 18, 2018, 15 pages.

Office Action for EP Application No. 16767369.8, dated Apr. 1, 2019, 7 pages.

Office Action for EP Application No. 16767369.8, dated Nov. 6, 2019, 8 pages.

Office Action for U.S. Appl. No. 14/248,155 dated Jan. 23, 2017, 14 pages.

Office Action for U.S. Appl. No. 14/248,155 dated Jun. 12, 2017, 15 pages.

Office Action for U.S. Appl. No. 14/248,155 dated Jun. 7, 2016, 13 pages.

Office Action for U.S. Appl. No. 14/635,192 dated May 4, 2017, 12 pages.

Office Action for U.S. Appl. No. 14/983,094 dated Feb. 8, 2018, 19 pages. .

Office Action for U.S. Appl. No. 14/983,094 dated Sep. 6, 2018, 20 pages.

Office Action for U.S. Appl. No. 16/195,607 dated Jun. 26, 2019, 6 pages.

Office Action from European Patent Application No. 15719292.3 dated Nov. 28, 2019, 13 pages.

Second Office Action for Australian Application No. 2016381225, dated May 10, 2019, 6 pages.

Stack Overflow, "Is there Windows system event on active window changed?" Nov. 15, 2015, retrieved from https://web.archive.org/web/20151115134250/https://stackoverflow.com/questions/4407631/is-there-wind . . . on Mar. 25, 2019, 1 page.

Wikipedia, "Concurrent Versions System," Jun. 20, 2013, retrieved from the Internet URL: http://en.wikipedia.org/w/index.php?title=Concurrent_Versions_System&oldid=560712674 on Sep. 23, 2014, 9 pages.

Wikipedia, "Operational Transformation," Last Modified Oct. 28, 2013, Retrieved from the Internet URL: http://en.wikipedia.org/wiki/Operational-transforma- tion on Jan. 6, 2014, 8 pages.

(56) **References Cited**

OTHER PUBLICATIONS

Wikipedia, "Rsync," Jun. 26, 2013, Retrieved from the internet URL:http://en.wikipedia.org/w/index.php?title=Rsync&oldid=561-668126 on Sep. 23, 2014, 8 pages.

Notice of Reasons for Refusal of Japanese Application No. 2019-029541, dated Mar. 19, 2020, 4 pages.

Non-Final Office Action from U.S. Appl. No. 16/560,901, dated Aug. 5, 2020, 7 pages.

Notice of Allowance from U.S. Appl. No. 15/867,693, dated Jul. 28, 2020, 9 pages.

Examination Report No. 3 for Australian Application No. 20190257465 dated Oct. 22, 2020, 5 pages.

Final Office Action from U.S. Appl. No. 16/510,018, dated Oct. 27, 2020, 15 pages.

Notice of Allowance from U.S. Appl. No. 16/257,503, dated Aug. 26, 2021, 2 pages.

Notice of Reasons for Refusal for Japanese Application No. 2020-113157 dated Jun. 14, 2021, 7 pages.

Non-Final Office Action from U.S. Appl. No. 16/280,845, dated Jun. 24, 2021, 22 pages.

Notice of Acceptance for Australian Application No. 2019284033 dated May 20, 2021, 3 pages.

Notice of Allowance from U.S. Appl. No. 17/460,164, dated Oct. 3, 2022, 10 pages.

Final Office Action from U.S. Appl. No. 17/501,680, dated Nov. 18, 2022, 10 pages.

Non-Final Office Action from U.S. Appl. No. 17/878,612, dated Dec. 29, 2022, 20 pages.

Final Office Action from U.S. Appl. No. 17/878,612, dated May 25, 2023, 21 pages.

Notice of Allowance from U.S. Appl. No. 17/501,680, dated Apr. 13, 2023, 8 pages.

Notice of Allowance from U.S. Appl. No. 17/501,680, dated Feb. 23, 2023, 08 pages.

Notice of Allowance from U.S. Appl. No. 17/501,680, dated Mar. 3, 2023, 02 pages.

Notice of Allowance from U.S. Appl. No. 17/878,612, dated Sep. 13, 2023, 6 pages.

* cited by examiner

Fig. 1

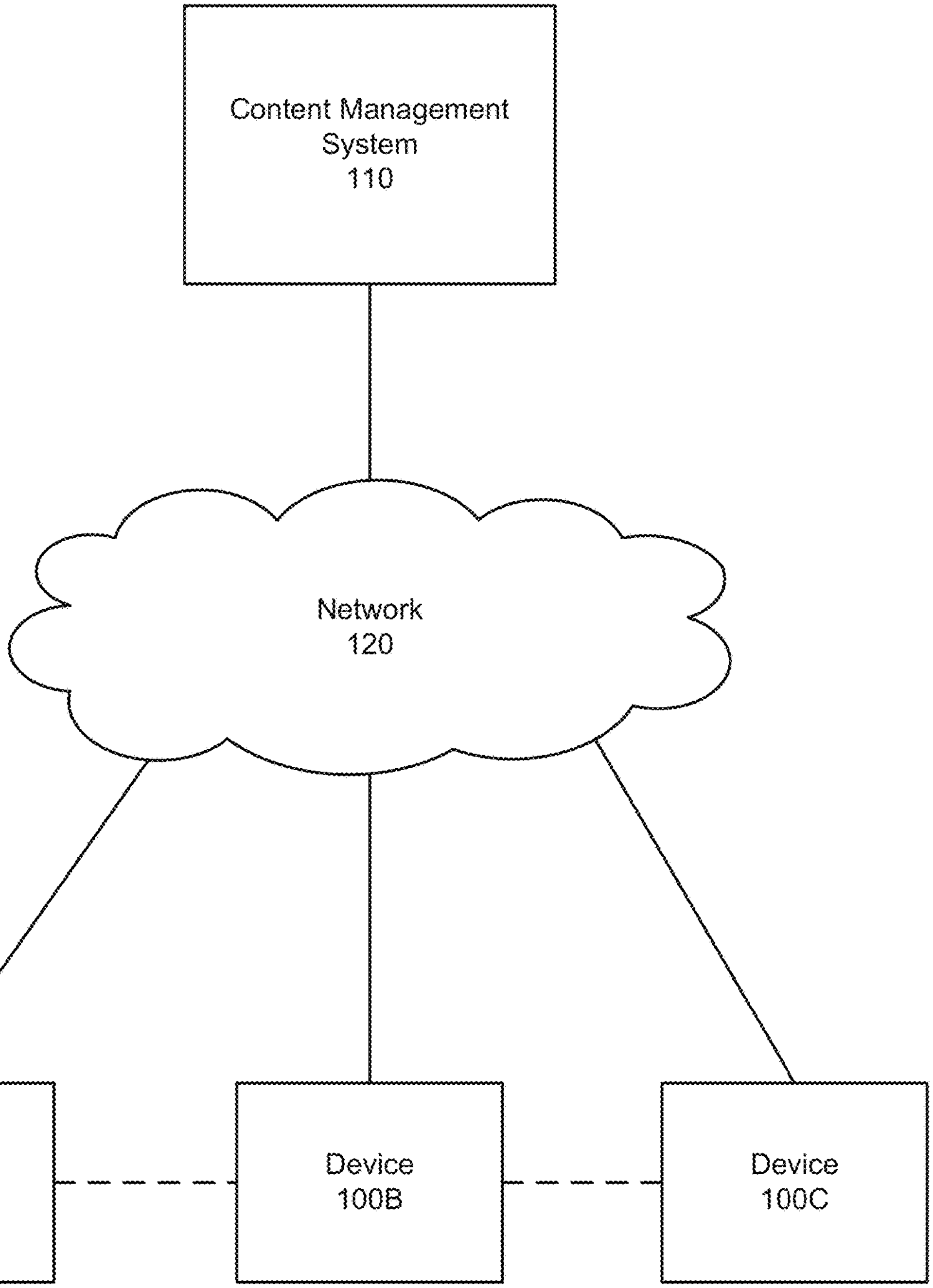

Content Management System
110
Network
120
Device
100A
Device
100B
Device
100C

Fig. 2

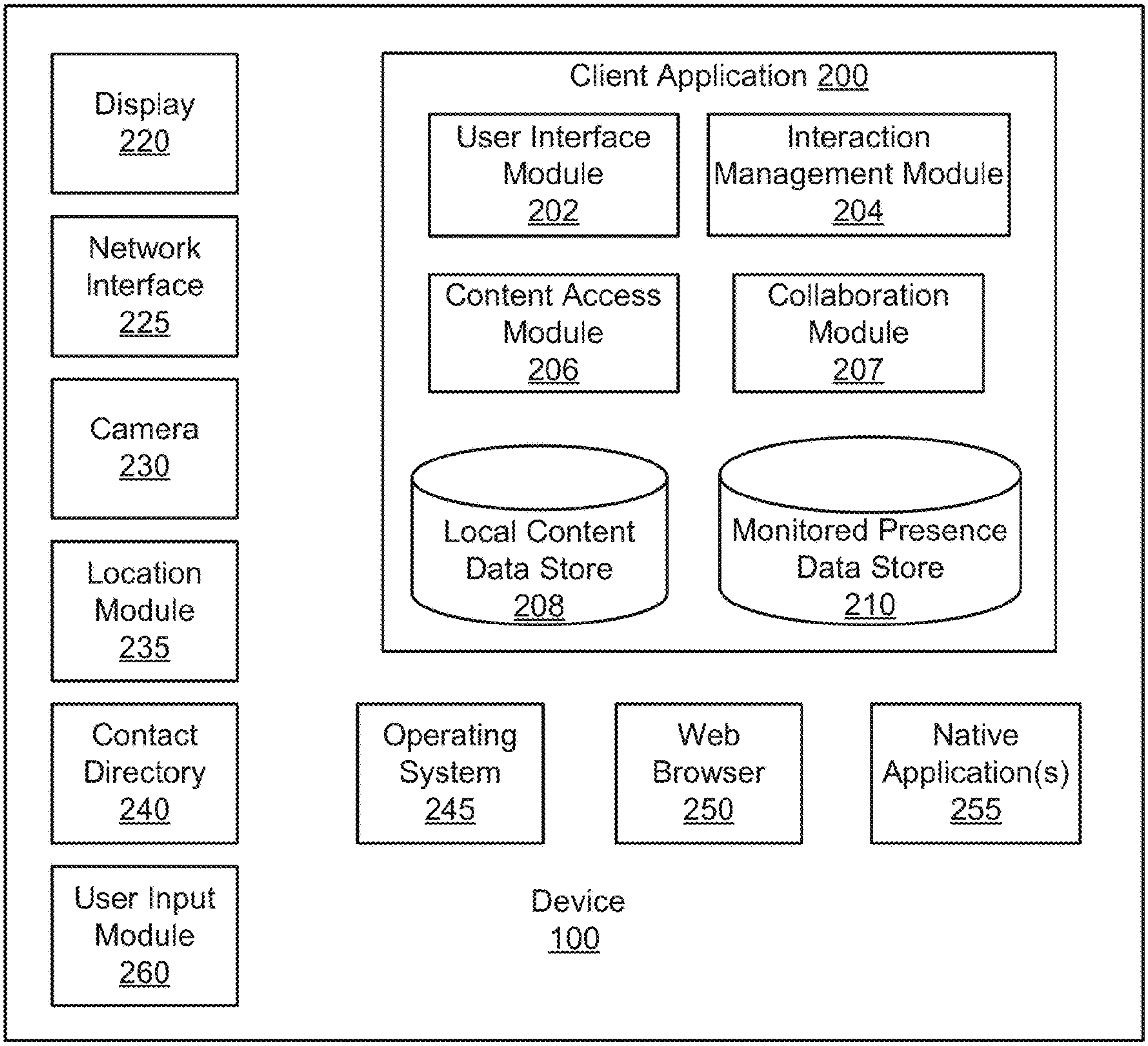

Display 220
Network Interface 225
Camera 230
Location Module 235
Contact Directory 240
User Input Module 260
Client Application 200
User Interface Module 202
Interaction Management Module 204
Content Access Module 206
Collaboration Module 207
Local Content Data Store 208
Monitored Presence Data Store 210
Operating System 245
Web Browser 250
Native Application(s) 255
Device 100

Fig. 3A

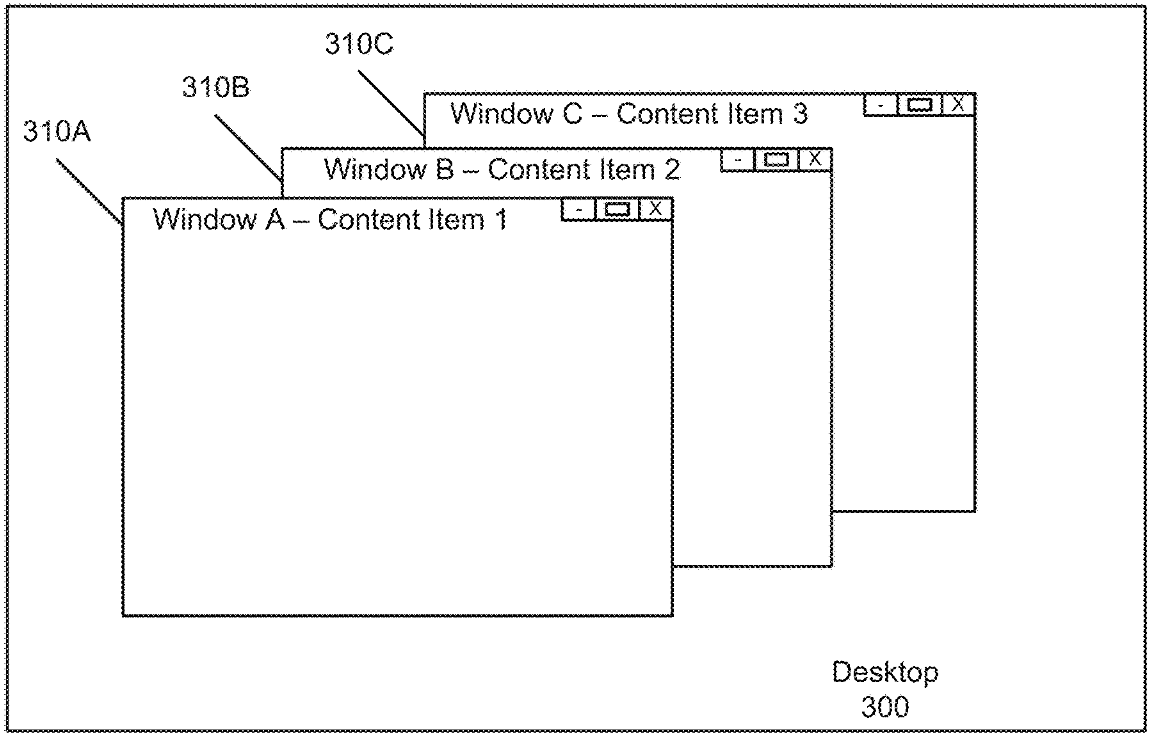
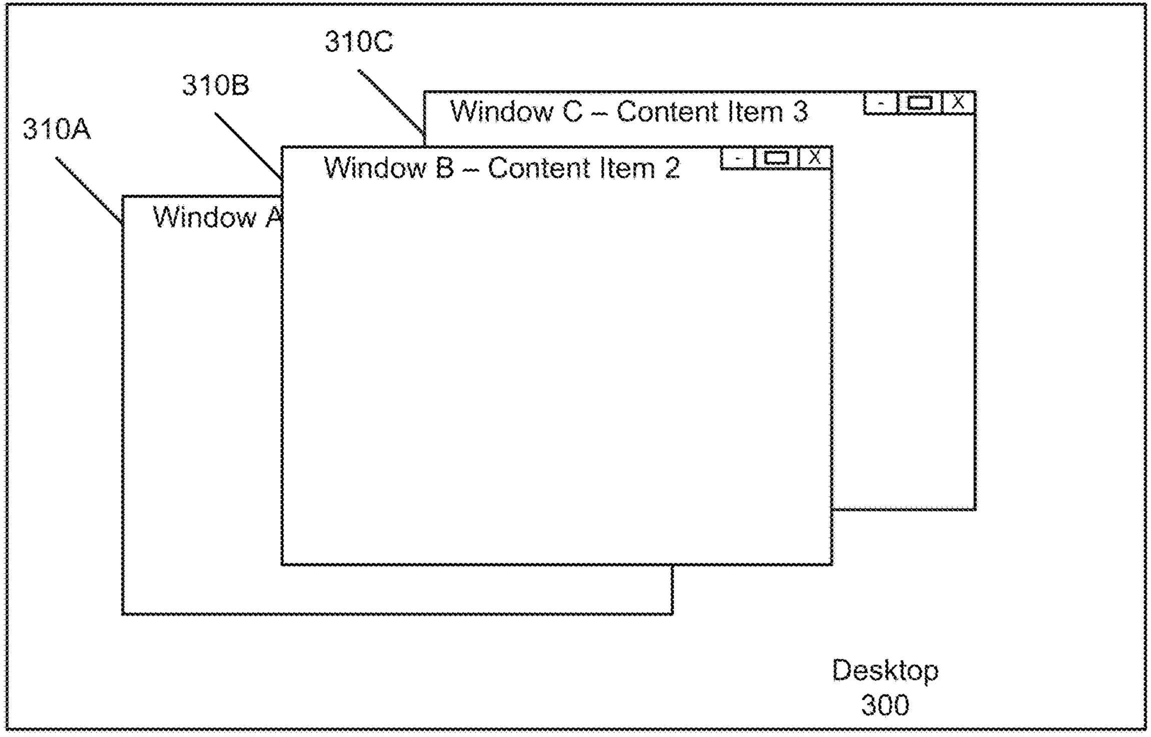

310C
310B
310A
Window C – Content Item 3
- ▭ X
Window B – Content Item 2
- ▭ X
Window A – Content Item 1
- ▭ X
Desktop
300

Fig. 3B


310C
310B
310A
Window C – Content Item 3
- ▭ X
Window B – Content Item 2
- ▭ X
Window A
Desktop
300

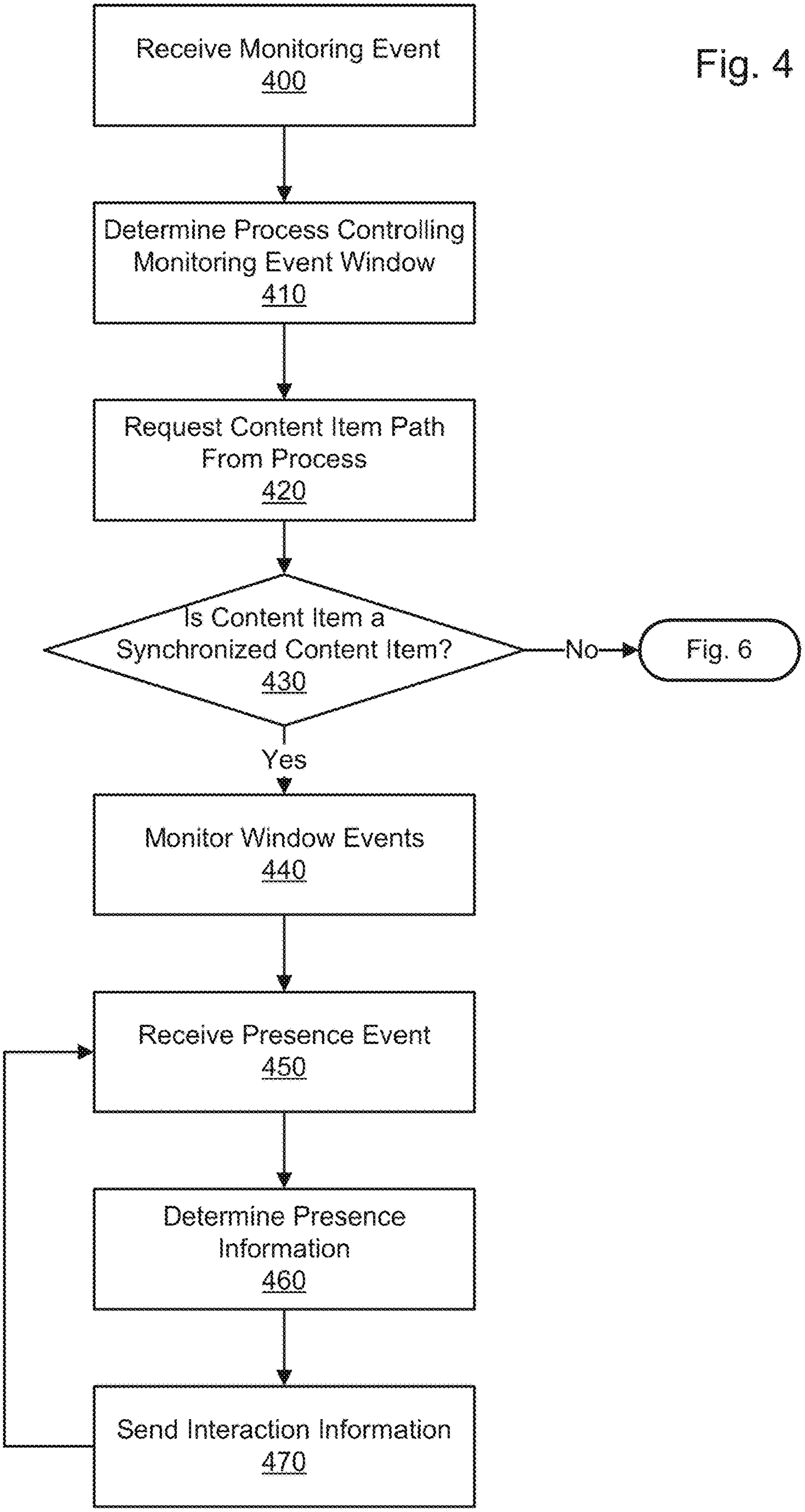
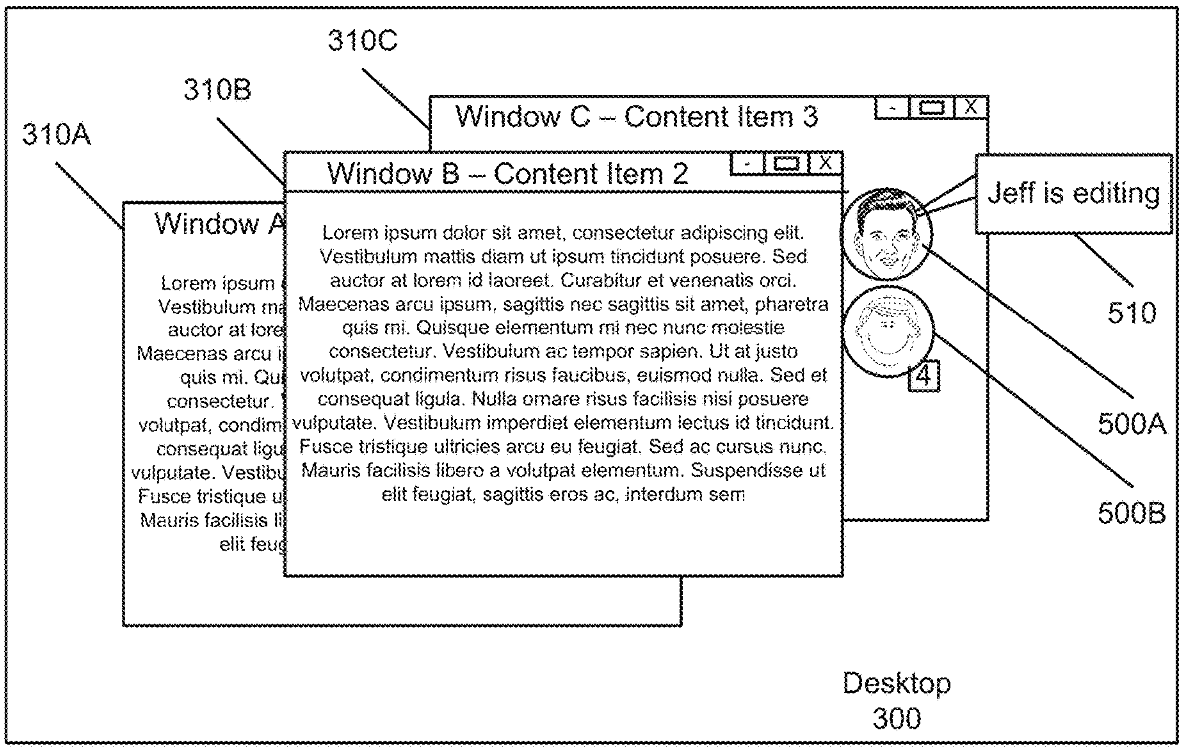

Receive Monitoring Event
400
Determine Process Controlling Monitoring Event Window
410
Request Content Item Path From Process
420
Is Content Item a Synchronized Content Item?
430
No
Fig. 6
Yes
Monitor Window Events
440
Receive Presence Event
450
Determine Presence Information
460
Send Interaction Information
470

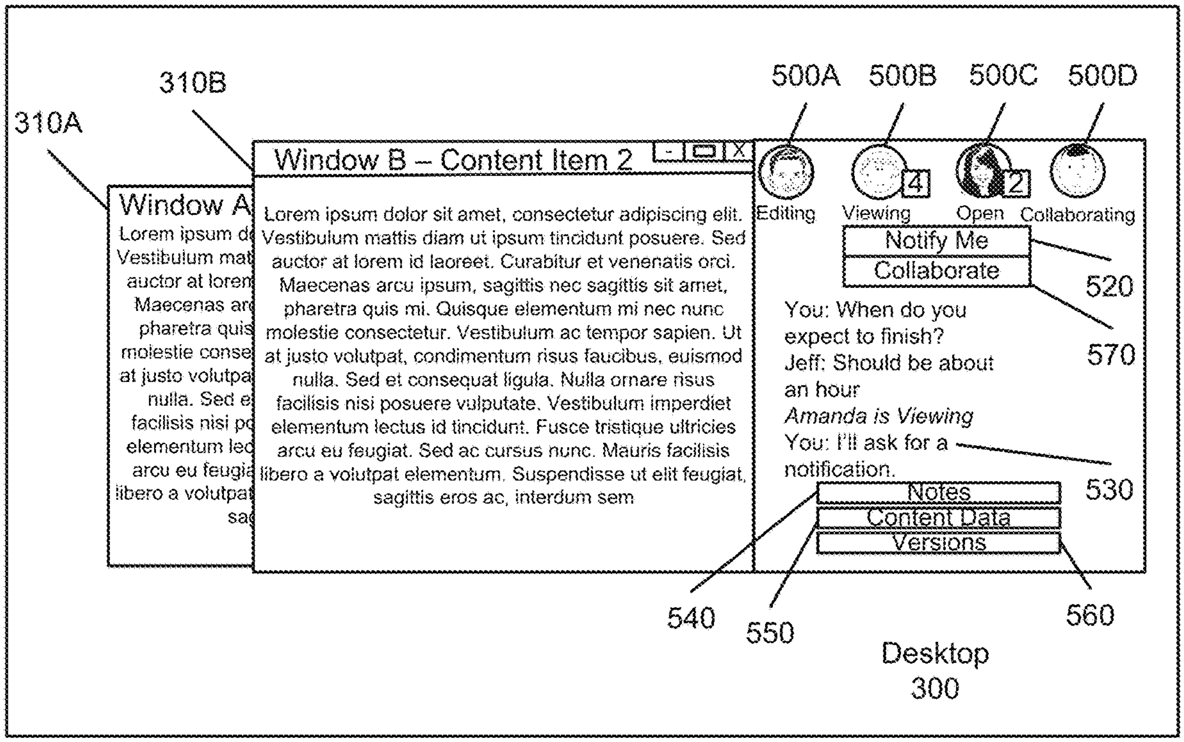
Fig. 4

Fig. 5A



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, pharetra quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

Fig. 5B

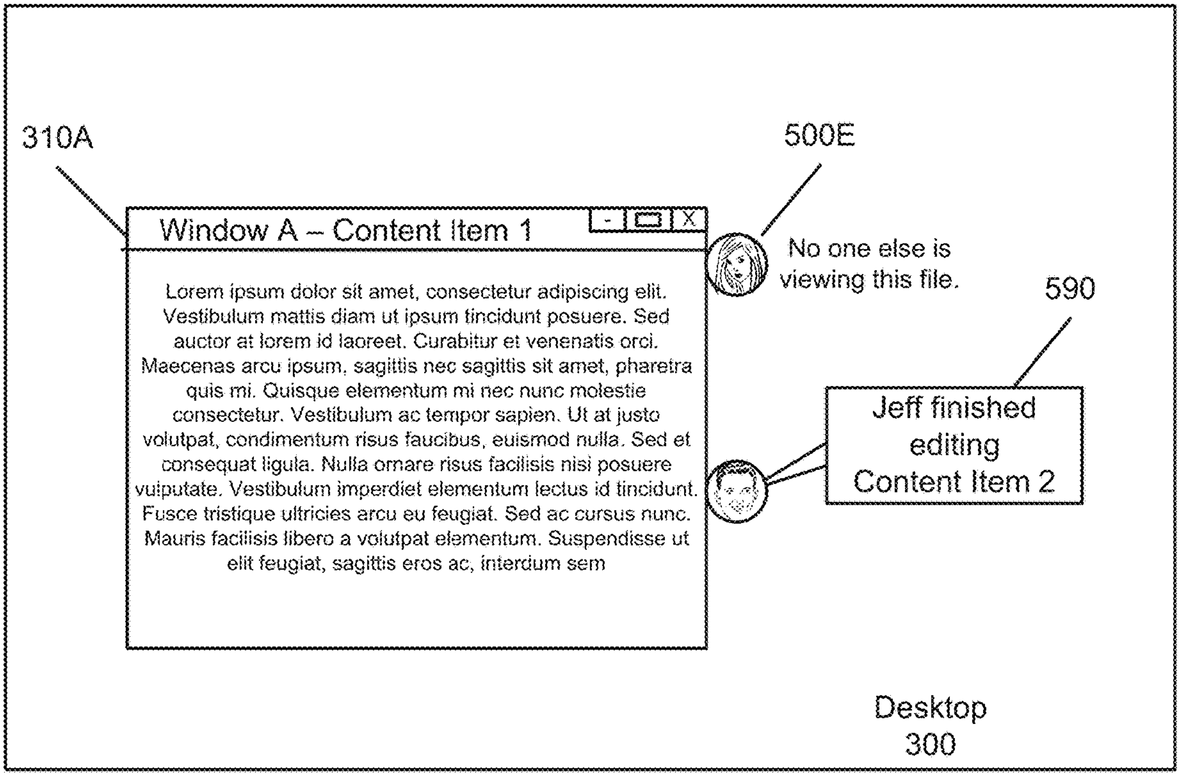


Window A
Lorem ipsum d
Vestibulum mat
auctor at lorem
Maecenas arc
pharetra quis
molestie conse
at justo volutpa
nulla. Sed e
facilisis nisi po
elementum lec
arcu eu feugia
libero a volutpat
sa

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, pharetra quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

You: When do you expect to finish?
Jeff: Should be about an hour
Amanda is Viewing
You: I'll ask for a notification.

Fig. 5C

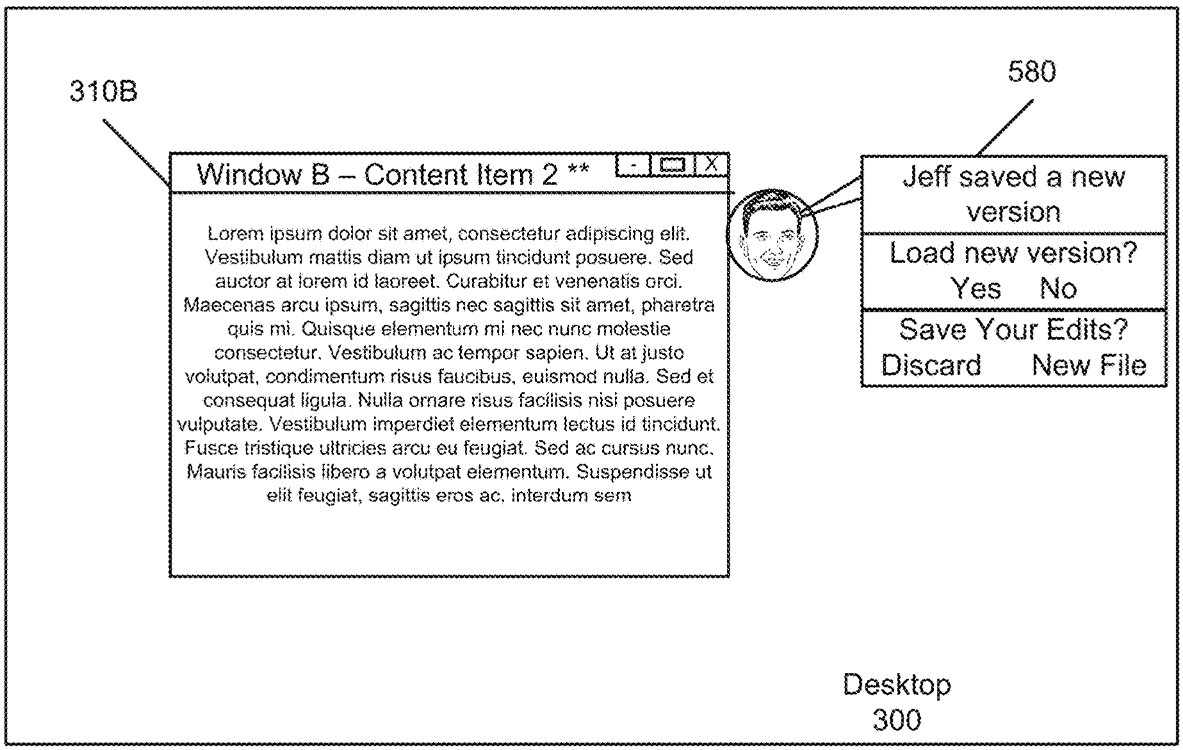


Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, pharetra quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

No one else is viewing this file.

Jeff finished editing Content Item 2

Fig. 5D



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, pharetra quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

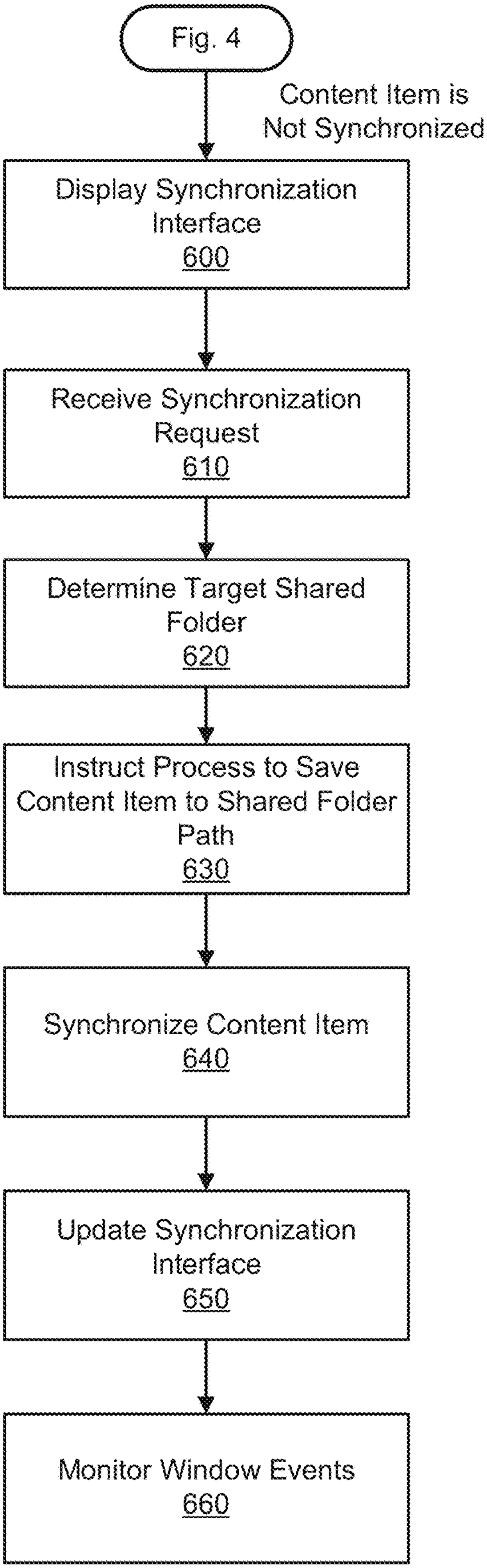
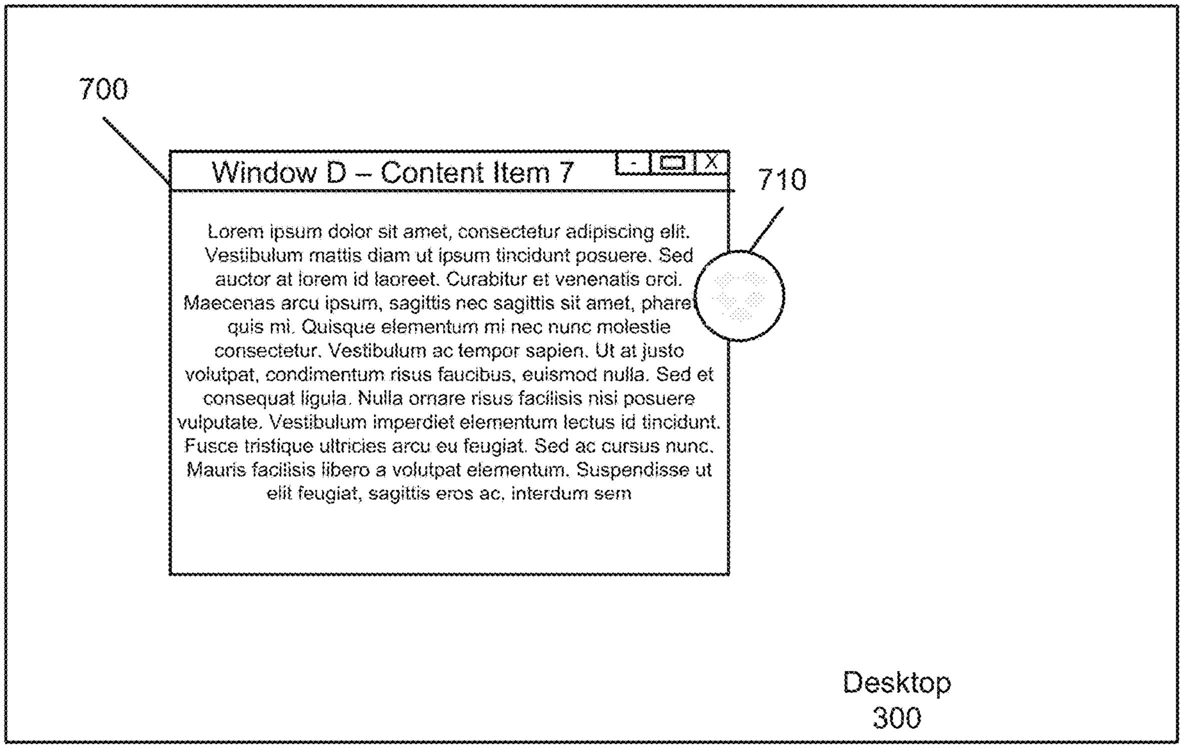


Fig. 4
Content Item is Not Synchronized
Display Synchronization Interface 600
Receive Synchronization Request 610
Determine Target Shared Folder 620
Instruct Process to Save Content Item to Shared Folder Path 630
Synchronize Content Item 640
Update Synchronization Interface 650
Monitor Window Events 660

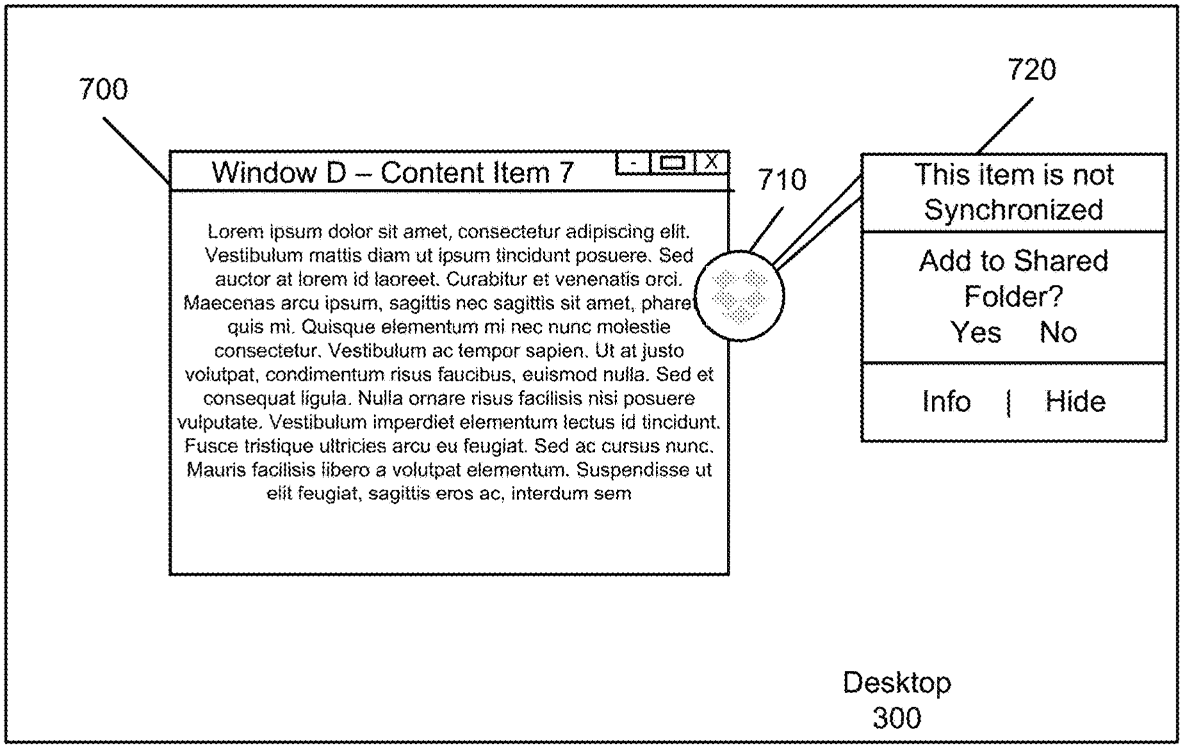
Fig. 6

Fig. 7A



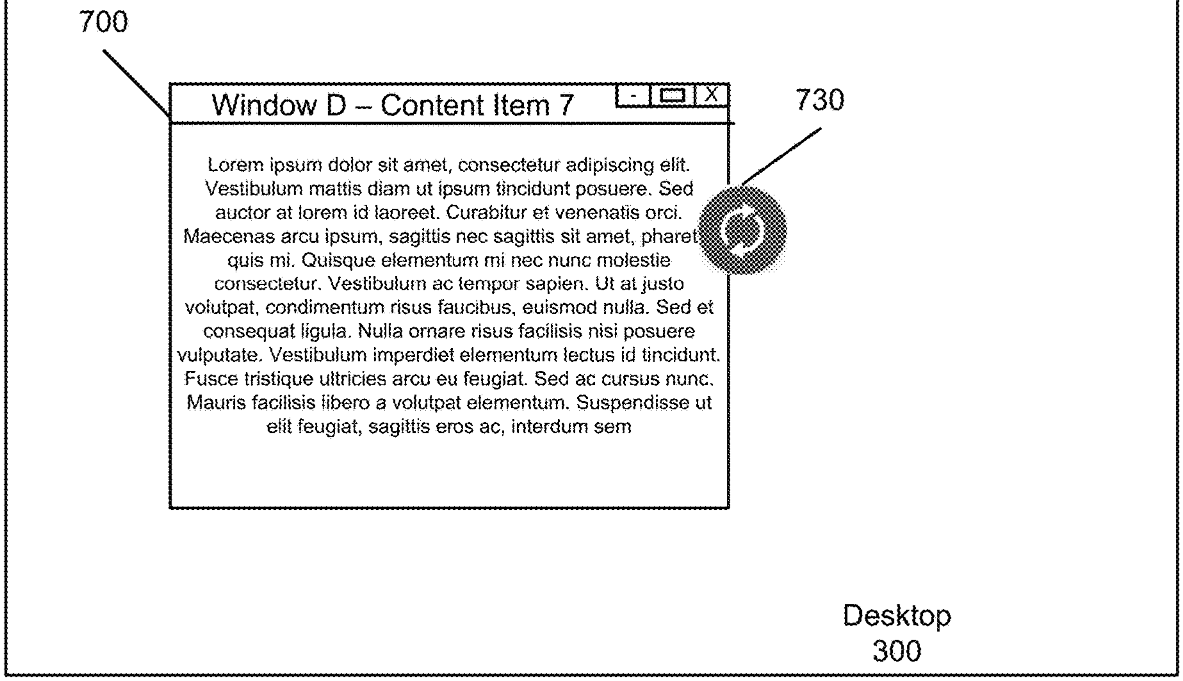
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, phare quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

Fig. 7B



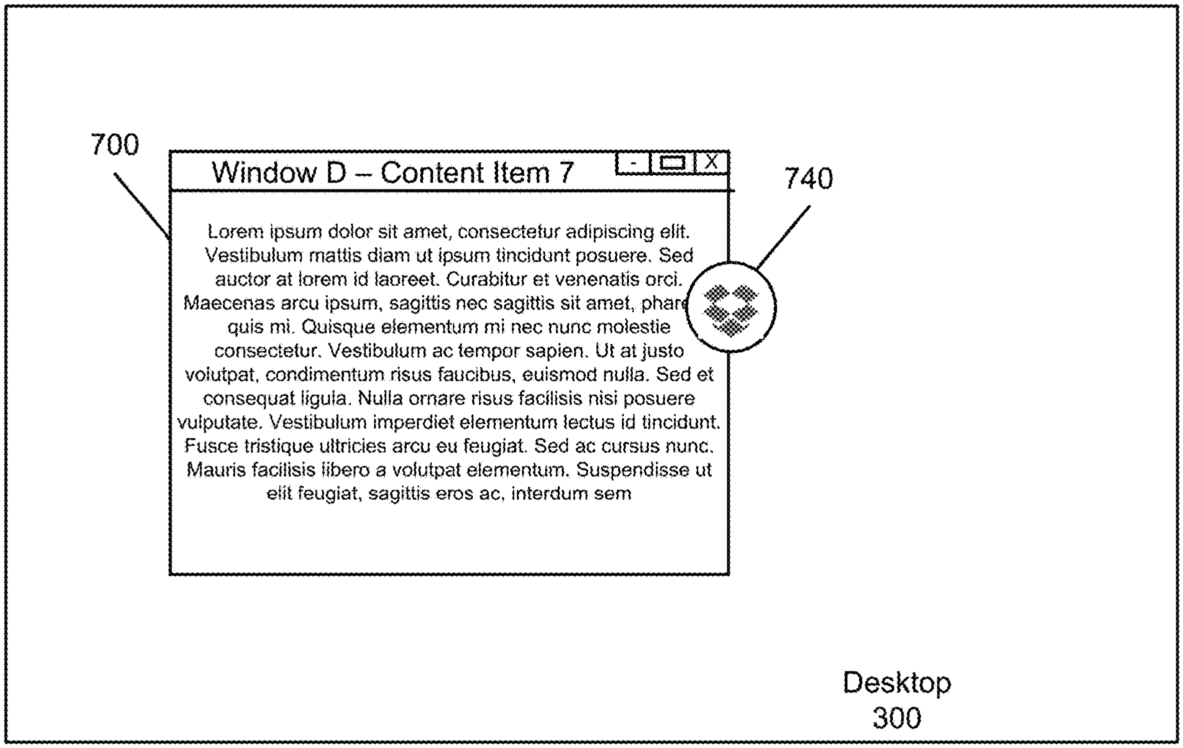
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, phare quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

Fig. 7C



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, pharet quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

Fig. 7D



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum mattis diam ut ipsum tincidunt posuere. Sed auctor at lorem id laoreet. Curabitur et venenatis orci. Maecenas arcu ipsum, sagittis nec sagittis sit amet, phare quis mi. Quisque elementum mi nec nunc molestie consectetur. Vestibulum ac tempor sapien. Ut at justo volutpat, condimentum risus faucibus, euismod nulla. Sed et consequat ligula. Nulla ornare risus facilisis nisi posuere vulputate. Vestibulum imperdiet elementum lectus id tincidunt. Fusce tristique ultricies arcu eu feugiat. Sed ac cursus nunc. Mauris facilisis libero a volutpat elementum. Suspendisse ut elit feugiat, sagittis eros ac, interdum sem

NATIVE APPLICATION COLLABORATION

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation of U.S. application Ser. No. 14/983,847, filed Dec. 30, 2015, now U.S. Pat. No. 10,620,811, which is incorporated by reference in its entirety.

BACKGROUND

This disclosure relates generally to sharing information among devices, and particularly to interactions between native applications and shared content.

Content management systems permit devices to synchronize content items with the content management system and other devices. A device stores a local copy of content items. When content items are added, deleted, and edited on a device, these modifications are sent to the content management system for storage and synchronization with other devices. To interact with a content item, users typically execute a native application on the device to view and modify the content item. When users seek to add a content item to a content management system, the user typically must exit the native application, and users do not have a convenient way to add content items to a shared folder while also viewing the content item in the native application.

SUMMARY

Described embodiments enable a set of users of devices sharing content items via a content management system to simultaneously view and modify a single instance of the same content item within a native application. In various embodiments, graphical and textual information is provided to users on their respective devices indicating that a shared content item is being viewed or edited on one or more other devices. In addition, in various embodiments users can collaboratively edit the shared content item from two or more devices.

A device in accordance with various embodiments stores a local copy of a shared content item, which is maintained and synchronized between devices. The synchronization may be performed by a content management system or by the devices among themselves. The device includes a native application that can be used to access the content item, such as a word processor, media viewer, media editor, and so forth. The native application displays information relating to the content item in a user interface element, such as a window. The device also includes a client application that monitors interactions with the content item and communicates information about those interactions to other devices sharing the content item either directly or via the content management system. The client application generates interaction information including data describing the user's interactions with a content item. The client application also facilitates collaboration between two or more devices.

Interaction information includes interactions with the client application and interactions with the native application. Interaction information may be determined from presence information, which include programmatic events that occur in the native application. Presence events include opening a content item, editing a content item, saving a content item, renaming a content item, moving a content item, and deleting a content item. Presence events also include interactions with the user interface element of the

native application, such as the user interface element gaining or losing focus. A focused user interface element is the user interface element that receives user inputs at the device. For example, if a user interface element has focus, presence information may be generated indicating that a user is "viewing" the content item via that user interface element. In one embodiment, the presence events are gathered by the client application, which is separate from the native application interacting with the content item. That is, in this embodiment the presence events are gathered by another application or process that is not integrated into the native application accessing the content item or the presence information.

The client application monitors events on the device to determine when a user is using a native application to interact with a content item. In one method of monitoring events, the client application registers with a process controlling a user interface element or the operating system to receive events related to the user interface element. The client application may monitor all such events, or only certain user interface elements associated with opening a content item. When a presence event occurs, the client application receives the presence event from the native application or from the operating system. The client application identifies a user interface element associated with the event and determines presence information associated with the event. The client application notifies the content management system for synchronized content items that the content item is being or has been interacted with, and sends the presence information to the content management system. The presence information (or other interaction information) may be sent directly to other devices.

The content management system receives the interaction information and may respond to the interaction information or send the interaction information to other devices based on the type of interaction information. Interaction information relating to messages is sent to other devices synchronized to the content item. Presence information is stored as a presence record, which in one embodiment indicates a content item, device, process, user interface element, and a type of presence (e.g., that the content item is open, viewed, being edited, or being collaboratively edited). A user presence may be determined that indicates a priority of interaction according to an ordering of interactions. In an example ordering, editing a content item has a higher priority than viewing the content item, which has a higher priority than opening the content item. The user presence describes a user's presence with respect to a content item and may be without reference to any particular device, process, or user interface element. The content management system may send the information to all devices that are synchronized with respect to the content item, or just to devices that are associated with an active presence record with respect to the content item. In one embodiment, devices register with the content management system to receive interaction information about a content item, which may specify a particular interaction to trigger notification. When the interaction occurs, the triggering interaction information is sent to devices associated with the notification.

When a monitoring event is received for a content item that is not synchronized to the content management system, a synchronization interface may be presented to the user. In the synchronization interface, the user may designate the content item to be synchronized to the content management system. When a request from the user is received to synchronize the content item, the client application determines a path for the content item in a shared folder and instructs the native application associated with the content item to save the content item to the shared folder. In this way, the native application may remain open with the content item during the saving and synchronizing process. When the content item is saved to a local shared folder, the client application synchronizes the content item to other devices or the content management system and during synchronization may update the synchronization interface to show the synchronization status of the content item. When the content item completes synchronization, the content item's status may be monitored normally, permitting sharing and other interactions with the content item via the content management system.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** shows an embodiment of an environment for content item synchronization including communication of interaction information and user collaboration

FIG. **2** shows various modules and components of a device in accordance with one embodiment.

FIGS. **3**A and **3**B show a user interface element focus change on a desktop display of a device.

FIG. **4** shows an example process for determining presence information associated with a content item according to one embodiment.

FIGS. **5**A-**5**D show example user interfaces displaying interaction information.

FIG. **6** shows an example process for adding a content item to a shared folder.

FIG. **7**A-**7**D show example user interfaces displaying a content item added to a shared folder.

The figures depict various embodiments of the present invention for purposes of illustration only. One skilled in the art will readily recognize from the following discussion that alternative embodiments of the structures and methods illustrated herein may be employed without departing from the principles of the invention described herein.

DETAILED DESCRIPTION

FIG. **1** shows an embodiment of an environment for content item synchronization including communication of interaction information and user collaboration. FIG. **1** includes devices **100**A, **100**B, **100**C (referred to generally as device **100**), content management system **110**, and network **120**. Three devices are shown only for purpose of illustration; in practice any number of devices may be present in the environment. Similarly, other modules or components described and illustrated throughout may include single or multiple instances as appropriate to the needs of the implementer and without loss of generality.

Device **100** may be any suitable computing device for locally storing and viewing content items and synchronizing the content items with content management system **110**. Examples of devices include desktop and laptop computers, hand-held mobile devices, tablet computers, and other computing devices. The operation of device **100** in various embodiments is further described below.

Each device **100** communicates with content management system **110** through network **120**. Network **120** is any suitable network and may include local networks, corporate networks, global networks, and any combination of these. In typical configurations, devices **100** communicate via a wired or wireless communication network to a local network service provider, and communicate with content management system **110** through the Internet. In certain configurations, devices **100**A, **100**B, and **100**C communicate directly with one another without network **120** as indicated in FIG. **1** by dashed lines. For example, devices **100** may communicate via a wired or wireless connection, such as wirelessly via a Bluetooth connection or a wired connection via a Universal Serial Bus (USB).

Content management system **110** provides content sharing and synchronization services for users of devices **100**. These services allow users to share content with users of other devices **100**. In addition to content sharing, content management system **110** updates shared content responsive to changes and enables synchronized changes to content items across multiple devices **100**. A user may synchronize content across multiple devices **100** owned by the user and associated with the user's account, and the user may share content that is synchronized with devices associated with other users' accounts. Content stored by content management system **110** can include any type of data, such as digital data, documents, media (e.g., images, photos, videos, audio, streaming content), data files and databases, source and object code, recordings, and any other type of data or file, collectively referred to here as "content items." Content items stored by content management system **110** may also be used to organize other content items, such as folders, tables, collections, albums, playlists, or in other database structures (e.g., object oriented, key/value etc.). In practice, various devices **100** may be synchronizing different groups of content items, based on user associations, permissions, content sharing permissions, and so forth. The operation of content management system **110** in various embodiments is further described below.

FIG. **2** shows various modules and components of device **100** in accordance with one embodiment. Device **100** includes display **220** for providing information to the user, and in certain client devices **100** includes a touchscreen. Device **100** also includes network interface **225** for communicating with content management system **110** via network **120**. Device **100** also includes a user input module **260**, which receives user inputs from various user input devices, such as a keyboard, a mouse, a trackpad, or other device. Other conventional components of a client device **100** that are not material are not shown, for example one or more computer processors, local fixed memory (RAM and ROM), as well as optionally removable memory (e.g., SD-card), power sources, and audio-video outputs.

Software modules include operating system **245** and one or more native applications **255**. Native applications **255** vary based on the client device, and may include various applications for creating, viewing, consuming, and modifying content stored on content management system **110**, such as word processors, spreadsheets, database management systems, code editors, image and video editors, e-book readers, audio and video players, and the like. Operating system **245** on each device provides a local file management system and executes the various software modules such as content management system client application **200** and native application **255**. A contact directory **240** stores information about the user's contacts, such as name, picture, telephone numbers, company, email addresses, physical address, website URLs, and the like. Further operation of native applications **255**, operating system **245**, and content management system client application **200** are described below.

In certain embodiments, device **100** includes additional components such as camera **230** and location module **235**. Camera **230** may be used to capture images or video for upload to the online content management system **110**. Location module **235** determines the location of device **100**, using for example a global positioning satellite signal, cellular tower triangulation, or other methods. Location module **235** may be used by client application **200** to obtain location data and add the location data to metadata about a content item, such as an image captured by camera **230**.

Client device **100** accesses content management system **110** in a variety of ways. Client application **200** can be a dedicated application or module that provides access to the services of content management system **110**, providing both user access to shared files through a user interface, as well as programmatic access for other applications. Client device **100** may also access content management system **110** through web browser **250**. As an alternative, client application **200** may integrate access to content management system **110** with the local file management system provided by operating system **245**. When access to content management system **110** is integrated in the local file management system, a file organization scheme maintained at content management system **110** is represented as a local file structure by operating system **245** in conjunction with client application **200**. Client application **200** may take various forms, such as a stand-alone application, an application plug-in, or a browser extension. Client application **200** includes user interface module **202**, interaction management module **204**, content access module **206**, local content data store **208**, monitored presence data store **210**, and collaboration module **207**.

In addition to handling other device tasks, operating system **245** displays information from applications executing on device **100** to a user via display **220**, which may include one or more user interface elements. Such user interface elements may vary based on the particular device and configuration. User interface elements include windows on a desktop interface as well as interface elements on a mobile device. Examples of operating systems that employ user interface elements such as windows are Microsoft Windows 8 by Microsoft Corporation of Redmond, Washington, and OS X by Apple Inc. of Cupertino, California. In addition, operating system **245** manages control of multiple native applications **255**, which may be executing simultaneously. The user interface elements may be layered, such that one layer overlaps another layer. In some operating systems and configurations, only a single user interface element is displayed at a given time. One user interface element is typically the active user interface element, meaning that it is the user interface element to which the operating system **245** routes user inputs, such as keyboard entry, cursor movement, touch sensors, touch gestures, and so forth. As understood by those of skill in the art, a window or other user interface element that is active at a particular time is often said to have focus. Users may select another user interface element to change the focus from one user interface element to another, and in some instances operating system **245** may change the focus without user input.

Typically, the user interface elements, e.g., windows, associated with native applications **255** are managed by operating system **245**, which maintains an association between process identifiers of executing native applications **255** and user interface element identifiers of the user interface elements. For example, a particular application may be associated with process id "2587", which may be managing multiple user interface elements, with user interface element identifiers **4**, **8**, and **10**. Each user interface element identifier may be separately associated with a particular content item opened by that native application **255**, and multiple user interface element identifiers and process identifiers may be associated with the same content item.

Operating system **245** also handles and recognizes various events. Such events include a request from native applications **255** to close or open a content item, a request from native applications **255** to close a window or other user interface element, and requests to change a user interface element focus, among many others. As described below, these events may be used by interaction management module **204** to recognize a change in presence related to a content item.

Client application **200** identifies interactions that take place with respect to a content item, such as when a user opens, closes or edits the content item on the device. These interactions are identified by client application **200** to generate interaction information describing the interaction with the content item. Interaction information includes interactions with client application **200** and interactions with native application **255**. Interaction information determined from actions of native application **255** is termed presence information. An application, such as client application **200** that determines interaction information and presence information is termed a presence application. Additional types of interaction information (in addition to presence information) include notes, messages, and notification requests related to the content item, which may be received by client application **200**. Messages may include chat messages to other devices, messages indicating a user's intent to interact with (e.g., to edit) a content item, and messages indicating a user's intent to begin a collaboration session. Notification requests may include a request to be notified when another user's interaction information changes. Interaction information also includes metadata modifications, such as versioning notes, or requests for further information stored at content management system **110** about the content item, such as a request to view versioning information or prior content item versions. Further examples of interaction information is described below.

This interaction information is transmitted to other devices **100** that are synchronized with respect to the content item. The indication of intent may for example alert a second user of the content item on another device that the first user would like to edit the content item, or initiatite a collaborative editing session. Client application **200** identifies when users are using a native application **255** to interact with a content item, and also receives chat or intent information from a user. In various embodiments, device **100** identifies a user's presence in a content item (i.e. that the user has the content item open or is editing the content item) through interaction with operating system **245** as described further below.

Device **100** receives content items from content management system **110** and permits users to view, modify, and interact with the content items using various native applications **255** stored on the device **100**. For example, device **100** may include a photo editing application that manipulates image content items, a word processing application that permits modification of text content items, or a computer-aided design (CAD) application that permits modification of drawing content items. As described further below, interaction information is determined by device **100** via user interactions applications and the interaction information is sent to other devices **100**. In addition, when device **100** receives interaction information relating to other devices **100**, the device **100** displays that interaction information.

In one embodiment, an application detecting interaction information relating to content items is distinct from the applications viewing or manipulating the content items. For example, the client application detecting interaction information is distinct from a photo editing application manipulating or displaying the image content items. In various embodiments, the application detecting interaction information is also responsible for synchronizing the content items with content management system **110**. Since the application detecting presence information may be distinct from the applications about which presence is detected, presence may be monitored for many applications and content items at once and without requiring integration of the presence monitoring into each type of content item viewer. That is, no special presence monitoring add-on or application modification is required, for example, for each of a photo editing application, a word processing application, and a playlist editing application.

FIGS. **3**A and **3**B show an example of a user interface element focus change on desktop **300** shown on display **220** of device **100**. In FIG. **3**A, windows **310**A, **310**B, and **310**C are displayed on desktop **300** and viewable by the user. In this embodiment, desktop **300** is a general container or frame maintained by operating system **245** that encloses user interface elements on display **220**. In FIGS. **3**A and **3**B, the user interface elements are windows **310** in a desktop computing environment. In other configurations, such as a mobile device, or other display with limited area, only a single user interface element might be displayed at a time. As shown by FIG. **3**A, window **310**A is the active window, shown as the front window, partially obscuring windows **310**B and **310**C. In FIG. **3**B, focus changed to window **310**B, which is now the front window and the active window. The focus may change due to user interaction with window **310**B, or due to a process requesting that its window become the active window. In certain operating systems and configurations, a user interface element has focus (e.g., is receiving user input) without being the front user interface element.

Referring again to FIG. **2**, to open a content item, native application **255** requests the content item from operating system **245** and receives a handle to the content item from operating system **245** for the content item. In some cases, application **245** does not maintain the handle, and may load the content item data into memory and subsequently close the content item handle even if native application **255** continues to use data from the content item or if the user enters edits to the content item. Accordingly, open content item handles are often not a reliable way to determine whether an application is interacting with a particular content item. As such, in certain embodiments, further behaviors exhibited by the native applications **255** are used to determine whether an application is editing a content item.

Native applications **255** also perform various behaviors when a user modifies a content item, and prior to the user saving the content item. These behaviors vary based on the application and operating system **245**. For example, some native applications **255** create a temporary content item with a filename that differs from the open content item, for example leading the temporary content item's filename with a tilde or other recognizable mark. In other examples, the native applications **255** changes the title of a user interface element associated with the content item, which may or may not be directly viewable by a user. In still further examples, native application **255** sets a flag indicating the content item has been modified. Native application **255** may also provide information regarding content item modification in response to a request from another application or the operating system. For example the Accessibility API in the OS X operating system as described above provides information regarding content items associated with a user interface element. Since an open content item handle may not reliably determine whether a content item is being edited by a native application **255**, these behaviors are used by presence management module **204** to determine presence relating to editing or modifying a content item as described further below.

Native applications **255** may typically be executed on device **100** independently from one another, and may permit communication between the applications and other applications or processes executing on device **100**. Native applications **255** typically provide information to processes using application programming interfaces (APIs), which permit applications to request information from the executing process. For example, native applications **255** may present an API permitting a request for user interface elements controlled by the application, or to indicate the title of a user interface element, or to request a path in a file system associated with a content item opened by the native application **255**. Similarly, operating system **245** may provide similar APIs to requesting processes, such as requesting information about a process that controls a particular user interface element.

Client application **200** manages access to content management system **110**. Client application **200** includes user interface module **202** that generates an interface to the content accessed by client application **200**, as variously illustrated herein, and is one means for performing this function. The generated interface is provided to the user by display **220**. Client application **200** may store content accessed from a content storage at content management system **110** in local content data store **208**. While represented here as within client application **200**, local content data store **208** may be stored with other data for client device **100** in non-volatile storage. When local content data store **208** is stored this way, the content is available to the user and other applications or modules, such as native application **255**, when client application **200** is not in communication with content management system **110**. Content access module **206** manages updates to local content data store **208** and uses synchronization logic to communicate with content management system **110** to synchronize content modified by client device **100** with content maintained on content management system **110**. One example of such synchronization is provided in U.S. patent application Ser. No. 14/040,584, filed Sep. 27, 2013 and is hereby incorporated by reference in its entirety. Client application **200** may take various forms, such as a stand-alone application, an application plug-in, or a browser extension.

Content management system **110** may provide additional data for synchronizing content items, such as information designating that a content item has been deleted, or that the device **100** may be viewing or editing an outdated version of a content item.

Interaction management module **204** obtains and manages interaction information relating to a user's synchronized content items. As described above, the interaction management module **204** is typically a distinct module from the native applications **255** being monitored by interaction management module **204** for presence information and executes as a separate process. Interaction management module **204** sends interaction information determined about a content item synchronized to device **100** to content management system **110**. Interaction management module **204** also receives interaction information relating to other users from content management system **110** for display to the user. As described further below, in one embodiment the interaction management module **204** displays presence information relating to other users by attaching a presence indicator to a user interface element associated with a synchronized content item. Further interaction information is also displayed with the presence indicator.

To determine many types of interaction information, interaction management module **204** receives interaction information through user interface elements, as further described below. To determine presence information related to a synchronized content item, interaction management module **204** monitors user interface elements associated with native applications **255**. Interaction management module **204** can monitor all user interface elements, or alternatively monitor just certain user interface elements after the user interface element is associated with a content item. Monitored presence data store **210** includes information maintained by interaction management module **204** to indicate that particular user interface elements are monitored to determine actions relating to a synchronized content item.

While shown here as a part of client application **200**, in various implementations the content access module **206** and interaction management module **204** are separated into distinct modules for performing their respective functions. Similarly, various modules and data stores are described separately throughout this disclosure for convenience and in various implementations may be combined or further separated into separate components as desired.

FIG. **4** shows an example process for determining presence information associated with a content item according to one embodiment. This process is typically performed by interaction management module **204**. Where the user interface elements are monitored only after being associated with a content item, interaction management module **204** uses events indicating that a content item is being opened by an application or user interface element to determine whether to monitor a user interface element. This is one example of an event that may associate a content item with a user interface element to initiate monitoring of the user interface element, termed a monitoring event. In other embodiments, a selection of user interface elements to monitor is determined in another way, or all user interface elements are monitored, in which case the interaction management module **204** may not use monitoring events. In another embodiment, the monitoring event includes a process saving a content item.

If enabled by operating system **245**, the interaction management module **204** may register with operating system **245** to receive monitoring events for specific applications. In these embodiments, operating system **245** notifies interaction management module **204** when a request to open or save a content item is received by operating system **245**. In this embodiment, interaction management module **204** receives a monitoring event **400** that indicates a window or other user interface element is interacting with a content item, which may be a synchronized content item (i.e., the process is interacting with the content item in a particular user interface element). The monitoring event designates at least a user interface element that triggered the monitoring event. In other embodiments, interaction management module **204** monitors events associated with user interface elements from time-to-time (e.g., five minute intervals) and queries whether the user interface elements are associated with any open content items. According to operating system **245** and native application **255** configuration, this query may be directed to operating system **245** or native application **255**. When a user interface element is associated with a newly opened content item, that newly opened content item is treated as a monitoring event to determine whether the newly opened content item is a content item synchronized with content management system **110** and that presence information should be determined for the newly opened content item.

When the monitoring event is received, interaction management module **204** determines **410** which process is responsible for the user interface element associated with the monitoring event. Interaction management module **204** typically determines the process by requesting the process ID associated with the user interface element from operating system **245**. In some embodiments, the interaction management module **204** identifies a process by requesting an identification of the process from the user interface element itself.

To confirm that the process and user interface element are correctly associated with one another and that the user interface element is still active, interaction management module **204** may also request from the process the identity of the currently active user interface element. The interaction management module **204** confirms that the currently active user interface element received from the process matches the user interface element associated with the monitoring event.

Using the process identifier, interaction management module **204** requests **420** any open content item from the process to obtain an associated directory path for the content item. The interaction management module **204** may designate the user interface element associated with the monitoring event with the request for the open content item's path. The interaction management module **204** requests the open item from the process or operating system using an interface available interface to the process or operating system. As one example, in the OS X operating system, the accessibility API may be used to access information relating to a content item and content item path for a user interface element, as known in the art. Using the content item path provided by the process, the interaction management module **204** determines whether the opened content item path corresponds to any synchronized content items. If so, interaction management module **204** determines that the content item accessed by the process is a content item synchronized to content management system **110** and associates that process and user interface element with the content item. In other embodiments, other methods may be used to determine whether a content item accessed by the process is a synchronized content item.

If the content item is synchronized **430** to content management system **110**, interaction management module **204** stores information relating to the content item, process, and user interface element, to monitor **440** the user interface element for events. When the content item associated with the monitoring event is not synchronized, the process may continue by displaying a synchronization interface to a user as further detailed in FIG. **6** and shown in FIGS. **7A-7D**. This monitoring information is stored in monitored presence data store **210**. To monitor and subsequently receive presence events related to the user interface element, interaction management module **204** registers to receive events associated with the user interface element. The registration process by the interaction management module **204** varies according to the configuration of device **100**. Typically the interaction management module **204** registers a request to receive presence events from operating system **245** or from the applicable process or user interface element. While the monitoring events determined whether a user interface element or process interacting with a synchronized content item, presence events are events that may indicate a change in state of a user's presence relating to the user interface element or process associated with a content item. Example presence events include a change in focus of a user interface element, closing a user interface element, closing a content item, opening a content item, and so forth based on the types of presence recognized by the interaction management module **204**. In various configurations, the presence events used by interaction management module **204** depend on the events operating system **245** and native application **255** make available for receipt by interaction management module **204**.

The presence events are used to determine presence information associated with the content item to which the presence event relates. For example, a presence event indicating that a user interface element that is associated with a content item has the focus will indicate that the user is viewing the content item, and hence the presence information for that content item indicates that state. Likewise, a presence event indicating that a user interface element unrelated to a content item has gained focus indicates that the content item associated with a previously focused user interface element has lost focus, and thus indicates that user is no longer be viewing the content item. Thus, presence information provides a level of semantic interpretation of the underlying presence event itself.

In addition to receiving presence events that the interaction management module **204** registered for, presence events may also be initiated by interaction management module **204** to confirm that presence information has not changed for a monitored user interface element. These presence events may be initiated if a threshold amount of time passed since the last presence event for a particular user interface element or process, or at particular intervals, e.g., every five minutes.

In addition to registering for events, interaction management module **204** may receive interaction events in other ways. In one embodiment, users may expressly indicate interaction information through a user interface element. The user interface element can be configured to allow the user to indicate, for example, that a user intends to revise a content item, to indicate that intent to other users who are editing or viewing the content item, for example by selection of a menu item or icon that represents the particular intent. The user interface element can also be configured to allow a user to indicate other intentions of the user, such as a user's intention to no longer view a content item, or to expressly indicate that a user is not or will not be present for a content item. Other users may use such "not present" intention to know that the content item is free for editing. User input interaction events may also include messages or chat features to be disseminated to other users associated with the content item, for example, to transmit a message to other users currently viewing the content item on other devices.

When a presence event is received **450**, interaction management module **204** determines whether any presence information has changed since the last presence event related to a monitored user interface element. For user-initiated interaction information, the interaction information may be the information provided by the user, for example the user's selection of a user interface element indicating that the user intends to modify a content item, or a user's chat message. For presence events, the interaction management module **204** queries the monitored process to determine the status of the monitored user interface element. In particular, the interaction management module **204** queries the process to determine if the monitored user interface element is the active user interface element. When the monitored user interface element is the active user interface element, the content item is being viewed by the user.

In some embodiments, in addition to detecting user presence with respect to a content item, interaction management module **204** also determines whether the content item is being or has been modified by the user. This further aspect enables presence information to be reported more granularly, for example with an indication that a user has a presence with respect to the content item as an editor rather than as a viewer. As the particular actions performed by applications when a content item is being modified may vary as described above, detecting one of these actions by interaction management module **204** indicates that the process has edited the content item. For example, according to the type of actions expected by the process when the content item is edited, interaction management module **204** may query the process to determine if the process indicates the content item has been flagged as modified, if the title information of the user interface element has changed, if a temporary file has been saved or cached, or any other data that suggests the content item has been modified. Interaction management module **204** may also query the operating system to determine if a content item has been saved that matches a temporary content item format, for example a content item with a filename similar to the content item, but with a tilde or other specialized variation of the filename. Such modifications indicate that the presence information associated with the content item should reflect that the user is editing the content item.

After determining **460** the presence information, any new presence information for a user interface element may be stored as monitored presence data store **250**. This presence information in one embodiment is stored on a user interface element-by-user interface element basis, such that multiple user interface elements by one process may be associated with the same content item, and have presence information individually managed. In one embodiment, presence information may change based on the current presence status. For example, when the presence information for a content item reflects that the content item is being edited, in one embodiment the presence for the content item in a user interface element is not changed when a user changes focus to another user interface element. Instead, the edited status is maintained with respect to that user interface element until a presence event indicates the user interface element is closed. In another embodiment, since editing has the potential to introduce modifications to the content item, in one embodiment the presence information for an edited document is not changed until the interaction management module **204** receives a notification that modifications to the content item are either committed or the modifications are discarded.

A content item with presence information indicating it is being viewed may have that status change when the user interface element loses focus, or within a threshold period of time of losing focus. This may be the case even if the user interface element associated with the content item is still open. In one embodiment, "viewed" presence information indicates whether a content item is associated with an active user interface element. In one embodiment, "viewed" presence information is retained until the user interface element is not active (or has lost focus) for longer than a threshold amount of time. In one embodiment, the content item is considered "viewed" while the content item is open by an application.

When there is a change to the interaction information, interaction management module **204** sends **470** the interaction information to content management system **110**. In one embodiment, the sent presence information includes an identifier of the content item, the process id, the user interface element id, and the presence status.

FIGS. **5**A-**5**D show example user interfaces displaying interaction information. Various user interface elements are similar to those depicted in FIGS. **3**A and **3**B. In FIG. **5**A, the example user interface displays presence information. To display presence information, interaction management module **204** provides presence indicator(s) **500** along a boundary or border of the window associated with the content item. In this example, interaction management module **204** receives presence information indicating that user Jeff is editing content item **2** and four users are viewing content item **2**. Interaction management module **204** displays presence indicator **500**A to indicate that Jeff is editing the document, presence indicator **500**B indicating users are viewing the document, presence indicator **500**C indicating users have the document open, and presence indicator **500**D indicating a collaborating user. In one embodiment, presence indicator **500** displays an icon or picture of a user associated with the presence indicator. The icon or picture may be obtained by interaction management module **204** from contact directory **240**, or may be received from content management system **110**. The individual user displayed by the presence indicator may vary depending on the embodiment, and may display, for example, the first user having a particular presence (e.g., the first viewer) or the user who most recently opened the content item.

Presence indicator **500** is displayed along with the window associated with the content item, and in one embodiment interaction management module **204** tracks the location of the window and displays presence indicator **500** adjacent to or near the window, for example alongside a border or boundary of the window. A supplemental presence indicator **510** may appear when a user hovers over presence indicator **500** to provide further information or interfaces for the user. In the example shown in FIG. **5**A, supplemental presence indicator **510** describes presence indicator **500**A, specifically that Jeff is associated with a presence of editing content item **2**. Supplemental presence indicator **510** may also appear when a presence changes, to indicate a new user is viewing or editing the document, for example.

The presence indicator **500** and supplemental presence indicator **510** may be located on any convenient area of display **220**. In one embodiment the presence indicator is displayed proximal the associated user interface element of the content item so as to visually indicate to the user the relationship between the presence indicator and the specific content item. In addition, the display of the presence indicator along a boundary or border of the window increases the likelihood that the user will notice the presence indicator. In one embodiment, the presence indicators **500** are displayed on or alongside a vertical edge of the window containing the content item (e.g., right edge as shown FIG. **5**A).

Alternatively, the presence indicator may be shown in a separate area of the display, such as a taskbar, or tray icon or may be a separate user interface element that does not interact with the user interface element of the content item. Though shown here as a single presence indicator **500** for each type of presence (editing or viewing), any number of presence indicators **500** may be shown related to the content item. For example, a circular indicator may include a count of users viewing the content item and another circular indicator may include a count of users editing the content item. Alternatively, where multiple presence indicators **500** are to be displayed, they may be ordered from top to bottom, where the ordering can be most recent to least recent, or highest priority to lowest priority, or a combination thereof (e.g., ordered by priority, and for presence indicators of the same priority, ordered by recency). The presence indicator, as shown, may also indicate an icon or picture associated with the other user. The indicators may also be color-coded to indicate the risk that a user will affect edits by other users. For example, the presence indicator may be red (or the presence indicator may turn red) when another user is editing, indicating to the user to coordinate any desired modifications with that user's changes. Likewise, the editing indicator may be yellow when other users are viewing the content item, and green when the current user is the only user viewing the content item.

FIG. **5**B shows a user interface through which a user may enter interaction information to the interaction management module **204**. This interface includes presence indicators **500**, in addition to further user interface elements. This interface may be presented in lieu of the example of FIG. **5**A, or may be presented as a supplemental presence indicator providing additional data regarding the content item.

The user interface of FIG. **5**B provides notification element **520**, permitting a user to indicate the user's desire to be notified when a presence change occurs related to another user. Upon selecting notification element **520**, the user may further select specific notification options, for example to be notified when a particular user's presence changes (e.g., Jeff is no longer editing), that all users have stopped a particular presence (e.g., no one is editing), or that no users have an active presence (e.g., that no one else is accessing content item **2**). The user's notification preference is sent to content management system **110** or the applicable users to request notification when the requested presence changes.

The interface also provides collaboration element **570**, permitting a user to initiate a collaboration session with other users who have the content item open. Upon selecting collaboration element **570**, the user may further select specific collaboration options. For example, choosing which users to include in a collaboration session and choosing whether to collaborate with editing control.

This interface also provides a chat interface **530** for users to communicate with other users present in the content item. The chat interface permits users to enter and receive messages to other users. Presence changes may also be indicated in the chat window, for example that Amanda is now viewing the content item. The chat window may permit users to specifically discuss information relating to that content item, such as when a user expects to finish editing the item. These chat messages are received by interaction management module **204** as interaction information and sent to other clients synchronized to the content item. This permits users to chat directly about a content item, even if the native application provides no chat functionality.

Notes user interface **540** permits a user to retrieve and enter notes stored in association with the content item. When notes interface **540** is selected, interaction management module **204** requests any notes or other persistent information from content management system **110** relating to the content item and displays any such notes to the user along with an interface for entering additional notes to be sent to other synchronized devices and content management system **110**. Like the chat window, this permits additional notes to be entered for a content item and application providing the user interface which may not natively provide for any note functionality. The notes element **540** may also be used when no other users are present within the content item and may be used to leave messages for other users about a content item.

Content data element **550** permits a user to request additional data about the content item, such as any related metadata, user permissions, and so forth. This permits a user to request details of the content item from the content management system directly from an interface near the user interface element associated with the content item.

Similarly, versions element **560** indicates a request for version information. Interaction management module **204** transmits the request for version information to content management system **110** as interaction information related to the content item. In response, content management system **110** identifies relevant version information for the content item. In one embodiment, a prior version is compared side-by-side with the version of the content item displayed in the user interface element. In another embodiment, the prior version is compared in-line in the user interface element.

FIG. **5**C shows a user interface in which a user is notified about a change in presence related to another content item. In this case, a user had previously selected to be notified when user Jeff finished editing content item **2**, using notification element **520**. After requesting notification, the user proceeded to close window **310**B relating to content item **2**, and is presently viewing content item **1**. In this embodiment, presence indicator **500**E indicates that the user is the only user with presence relating to this content item. This may be espressly indicated ("No one else is viewing this file") as shown, or may be provided if a user interacts with presence indicator **500**E. Interaction management module **204** receives a notification that Jeff has finished editing content item **2**. The notification **590** is displayed to the user. This notification is provided though the user may be interacting with another application or another content item than the content item to which the notification relates.

FIG. **5**D shows a user interface in which a user receives a notification that a content item has been modified. In this example, a new version of content item **2** viewed by the user has been saved by another user, Jeff. Interaction management module **204** receives a message from content management system **110** indicating that content item **2** has been modified and a new version is available. Typically, the user's edits may be lost if the user attempted to synchronize the user's edits to content management system **110** because the user's edits would be out of date with the version maintained by content management system **110**. Using the presence information indicating the user is editing the content item, the user can be provided choices to assist in incorporating any edits from Jeff into the user's edits. A version notification **580** is presented to the user, along with choices to make in response to the new version. In this example, the user may elect to load the new version, discard the user's edits, or save the user's edits as a new file. The particular choices available to a user may be provided by content management system **110** based on content management system **110**'s record of the device's presence **100**, as further described below. In another embodiment, interaction management module **204** determines options for display to the user.

In embodiments where interaction management module **204** is a separate application from the native application **255** of window **310**B, presence indicators **500** are provided by interaction management module **204**. Thus, interaction management module **204** monitors presence information associated with the application, and displays presence information relating to other users for a content item associated with

the application. When other windows are activated, the same interaction management module **204** displays presence information relating to the activated window.

To display the presence indicator(s), interaction management module **204** receives the presence information for other devices, typically from the content management system **110**. Interaction management module **204** determines that a synchronized content item is in use by a window or process, and that the window is being displayed to the user. For example, interaction management module **204** may receive a presence event from the window or process being monitored as described above. When the presence event indicates that the window has the focus, interaction management module **204** adds the presence indicator to the display. Interaction management module **204** in one embodiment also tracks movement of the window in the desktop and moves the presence indicators to maintain the location of the presence indicators relative to the window.

In one embodiment, to add the presence indicator to the display, interaction management module **204** determines the location of the focused window and its boundaries, and adds the presence indicator adjacent to the window boundary. When interaction management module **204** identifies that the focused window has changed, the presence indicator for that content item may also be removed until that window is focused again.

Referring again to FIG. **5**B, in one embodiment, collaboration element **570** is displayed in association with a content item that is open for viewing or editing in native applications on at least two devices. A viewing or editing user may initiate collaboration via collaboration element **570**. During collaboration one device assumes the role of primary device, and other device(s) are secondary devices. The primary device is the device having the instance of the content item in which collaboration occurs. Secondary devices therefore provide viewing and editing functionality with respect to the instance of the content item located on the primary device. There may be multiple secondary devices for a given collaboration session, but only one primary device.

Prior to collaboration, an instance of the content item may be open for viewing or editing on what will become the secondary device. In one embodiment, collaboration element **570** prompts one of the users, e.g., the requesting user, to choose which device will be the primary device for the collaboration session. In another embodiment, the primary device is chosen by collaboration module **207**. For example, if one user has the content item open for editing and another has the item open for viewing, collaboration module **207** designates the device on which the item is being edited as primary.

FIG. **6** shows an example process for adding a content item to a shared folder. As discussed with respect to FIG. **4**, FIG. **6** continues a process when a monitoring event is received for a content item that is not synchronized with the content management system. The process shown in FIG. **6** may be performed by client application **200**. In this example, the monitoring event may be a content item opened or saved by a process associated with a window or other user interface element of a native application. In some examples, the synchronization interface is displayed for any content item of a native application that is not synchronized with a shared folder. In other examples, the synchronization interface is displayed after a threshold amount of time that the content item is saved, such as five or ten seconds after saving. A synchronization interface is displayed **600** to adjacent to or near the window of the native application, and may overlay a portion of the window. Example user interfaces for the synchronization interface are shown in FIGS. **7**A-**7**D. The synchronization interface permits a user to indicate a request to add the content item to a shared folder synchronized with content management system **110** or one or more other client devices **100**. Next the user may provide a request to add the content item to the shared folder, which is received **610** by client application **200**.

To save the content item to a shared folder, client application **200** determines **620** a target shared folder for the content item. That is, the client application determines the location to save the content item, such as a specific shared folder or other location or collection in local content data store **208** that is synchronized with content management system **100** or other client devices **110**. In one embodiment, the synchronization interface displays a file browser interface for a user to select a location in local content data store **208** for storing the content item. Alternatively, the target shared folder may be the automatically selected, such as a root of a directory tree for synchronized content items, or another automatically selected location.

When a user requests to add the content item to the shared folder, the content item is being accessed by the native application and the native application may have a handle or other file access association with the content item. To move the content item to the target shared shared folder, the client application **200** instructs **630** the process to save the content item to the target shared folder. The instruction to the process may be provided, for example, via an application programming interface, and specify a window of the content item and a path for the shared folder. In another example, rather than instruct the process to save the content item, the client application **200** instructs the process to close the content item at its existing location. The client application **200** copies the content item from its existing location to the target shared folder, and instructs the process to open the content item at the target shared folder. By moving the content item to the shared folder in this way, the content item may continue to be used by the user in the native application and the content item may be seamlessly added to the shared folder without interrupting normal use of the content item in the native application.

When the content item is saved to the target shared folder, a copy of the content item may still be stored at the prior location of the content item. The client application may delete the copy of the content item, or may leave the copy of the content item at the prior location.

When the content item is saved to the target shared folder, the content item is synchronized **640** with other systems, such as client devices **110** and content management system **100** synchronizing that shared folder. While the content item is being synchronized, the synchronization interface is updated **650** to indicate that the content item is being synchronized. When the synchronization completes, the synchronization interface may be updated again to indicate that the content item has successfully been synchronized. At that point, additional options for interacting with the content item with respect to the synchronization may be available, such as sharing presence information with other users for the content item, sharing the content item to other users, and other uses of synchronized content items. To perform additional presence actions for the content item, the window events may subsequently be monitored **660** as discussed with respect to FIG. **4**.

FIG. **7**A-**7**D show example user interfaces displaying a content item added to a shared folder. These user interfaces may be generated, for example, by user interface module **202**, and is one means for doing so. In this example, window **700** displays a content item, here "content item **7**." This content item is not synchronized with a shared folder, and as described above, a synchronization interface **710** is displayed near the window **700**. Synchronization interface **710** is shown in this example along a side portion of the window, though synchronization interface **710** may not overlap with the window **700** in some embodiments, or may dock to a side portion of the window. User interface module **202** may monitor the location of window **700** to place synchronization interface **710** at a position relative to window **700**.

In the example shown in FIG. **7**A, synchronization interface **710** displays an icon indicating that the content item is not a part of a shared folder. In this example, synchronization interface **710** displayed a grayed-out icon to represent that the content item is not shared, though other icons or indications may be used.

When a user interacts with synchronization interface **710**, synchronization menu **720** may be displayed as shown in FIG. **7**B. The user's interaction may include, for example, a click or selection of the synchronization interface **710**, or may include a cursor maintained over the location of location interface **710** for a threshold period of time (e.g., "hovering"), or other interactions.

Synchronization menu **720** provides information to the user about the content item, such as "This item is not synchronized" and provides an option for the user to add the content item to the shared folder. When the user selects "yes" to add the content item to the shared folder, the content item may be added to the shared folder as described with respect to FIG. **6**. In adition, an interface element may be provided (not shown) for a user to select the target shared folder to which to save the content item. When a user selects "No" the content item is not added to a shared folder, and synchronization menu **720** is closed. Sychronization menu **720** in this example also provides interfaces for additional information or to hide synchronization menu **720**. When the user interacts with the interface for additional information ("Info"), additional information is displayed to the user, for example the advantages and features of synchronizing the content item. When the user interacts with the interface to "Hide," the synchronization menu **720** is closed, and the synchronization interface **710** also removed for this content item.

After the user selects "yes" and the content item is added to the shared folder, the synchronization menu **720** in this example is closed, and the closed, and the synchronization interface **710** changes to indicate a synchronization of the content item as shown in synchronization interface **730** in FIG. **7**C. In this example, synchronization interface **730** displays circular arrows to indicate synchronization of the content item to other client devices **110** or content management system **100**.

After the content item is synchronized, the synchronization interface is changed as shown in FIG. **7**D. Synchronization interface **740** indicates that the content item is synchronized with other shared folders, and provides an indication with the native application that the content item has been successfully added to the shared folder and is synchronized with other devices. A user may further interact with synchronization interface **740**, to access synchronization features for the content item, such as to provide presence information, versioning, notes, and other features of synchronized content provided by client application **200**.

The foregoing description of the embodiments of the invention has been presented for the purpose of illustration; it is not intended to be exhaustive or to limit the invention to the precise forms disclosed. Persons skilled in the relevant art can appreciate that many modifications and variations are possible in light of the above disclosure.

Some portions of this description describe the embodiments of the invention in terms of algorithms and symbolic representations of operations on information. These algorithmic descriptions and representations are commonly used by those skilled in the data processing arts to convey the substance of their work effectively to others skilled in the art. These operations, while described functionally, computationally, or logically, are understood to be implemented by computer programs or equivalent electrical circuits, microcode, or the like. Furthermore, it has also proven convenient at times, to refer to these arrangements of operations as modules, without loss of generality. The described operations and their associated modules may be embodied in software, firmware, hardware, or any combinations thereof.

Any of the steps, operations, or processes described herein may be performed or implemented with one or more hardware or software modules, alone or in combination with other devices. In one embodiment, a software module is implemented with a computer program product comprising a computer-readable medium containing computer program code, which can be executed by a computer processor for performing any or all of the steps, operations, or processes described.

Embodiments of the invention may also relate to an apparatus for performing the operations herein. This apparatus may be specially constructed for the required purposes, and/or it may comprise a general-purpose computing device selectively activated or reconfigured by a computer program stored in the computer. Such a computer program may be stored in a non-transitory, tangible computer readable storage medium, or any type of media suitable for storing electronic instructions, which may be coupled to a computer system bus. Furthermore, any computing systems referred to in the specification may include a single processor or may be architectures employing multiple processor designs for increased computing capability.

Embodiments of the invention may also relate to a product that is produced by a computing process described herein. Such a product may comprise information resulting from a computing process, where the information is stored on a non-transitory, tangible computer readable storage medium and may include any embodiment of a computer program product or other data combination described herein.

Finally, the language used in the specification has been principally selected for readability and instructional purposes, and it may not have been selected to delineate or circumscribe the inventive subject matter. It is therefore intended that the scope of the invention be limited not by this detailed description, but rather by any claims that issue on an application based hereon. Accordingly, the disclosure of the embodiments of the invention is intended to be illustrative, but not limiting, of the scope of the invention, which is set forth in the following claims.

What is claimed is:

1. A method comprising:

detecting, by a client application installed on a client device, a change in state of a user's presence on a content item controlled by a native application installed on the client device, wherein the client application is not integrated into the native application;

determining, by the client application, whether the content item is saved to a local folder of the client device as opposed to a synchronized folder synchronized with a content management system, wherein the local folder is not synchronized to the content management system that is remote to the client device;

responsive to determining, by the client application, that the content item is saved to the local folder of the client device as opposed to the synchronized folder synchronized with the content management system:

generating for display, by the client application, a synchronization interface comprising a selectable option for a user to request synchronization of the content item to the content management system;

receiving a selection of the selectable option to synchronize the content item from the synchronization interface; and

in response to receiving the selection of the selectable option, synchronizing the content item with the content management system by moving the content item from the local folder to the synchronized folder.

2. The method of claim **1**, wherein synchronizing the content item with the content management system comprises synchronizing the content item to one or more shared devices.

3. The method of claim **1**, further comprising identifying a location of the synchronized folder.

4. The method of claim **3**, wherein the location of the synchronized folder is specified by the user.

5. The method of claim **3**, wherein the location of the synchronized folder is a root of a directory tree for synchronized content items.

6. The method of claim **1**, wherein detecting the change in state comprises detecting a saving operation with respect to the content item.

7. The method of claim **1**, wherein detecting the change in state comprises detecting that the content item has been opened.

8. A non-transitory computer-readable medium comprising instructions executable by one or more processors of a system, the instructions, when executed by the one or more processors, causing the one or more processors to perform operations, the instructions comprising instructions to:

detect, by a client application installed on a client device, a change in state of a user's presence on a content item controlled by a native application installed on the client device, wherein the client application is not integrated into the native application;

determine, by the client application, whether the content item is saved to a local folder of the client device as opposed to a synchronized folder synchronized with a content management system, wherein the local folder is not synchronized to the content management system that is remote to the client device;

responsive to determining, by the client application, that the content item is saved to the local folder of the client device as opposed to the synchronized folder synchronized with the content management system:

generate for display, by the client application, a synchronization interface comprising a selectable option for a user to request synchronization of the content item to the content management system;

receive a selection of the selectable option to synchronize the content item from the synchronization interface; and

in response to receiving the selection of the selectable option, synchronize the content item with the content management system by moving the content item from the local folder to the synchronized folder.

9. The non-transitory computer-readable medium of claim **8**, wherein the instructions to synchronize the content item with the content management system comprise instructions to synchronize the content item to one or more shared devices.

10. The non-transitory computer-readable medium of claim **8**, wherein the instructions further comprise instructions to identify a location of the synchronized folder.

11. The non-transitory computer-readable medium of claim **10**, wherein the location of the synchronized folder is specified by the user.

12. The non-transitory computer-readable medium of claim **10**, wherein the location of the synchronized folder is a root of a directory tree for synchronized content items.

13. The non-transitory computer-readable medium of claim **8**, wherein the instructions to detect the change in state comprise instructions to detect a saving operation with respect to the content item.

14. The non-transitory computer-readable medium of claim **8**, wherein the instructions to detect the change in state comprise instructions to detect that the content item has been opened.

15. A system comprising:

one or more processors configured to execute instructions;

a non-transitory computer-readable medium comprises instructions, which when executed by the one or more processors, cause the system to perform operations comprising:

detecting, by a client application installed on a client device, a change in state of a user's presence on a content item controlled by a native application installed on the client device, wherein the client application is not integrated into the native application;

determining, by the client application, whether the content item is saved to a local folder of the client device as opposed to a synchronized folder synchronized with a content management system, wherein the local folder is not synchronized to the content management system that is remote to the client device;

responsive to determining, by the client application, that the content item is saved to the local folder of the client device as opposed to the synchronized folder synchronized with the content management system:

generating for display, by the client application, a synchronization interface comprising a selectable option for a user to request synchronization of the content item to the content management system;

receiving a selection of the selectable option to synchronize the content item from the synchronization interface; and

in response to receiving the selection of the selectable option, synchronizing the content item with the content management system by moving the content item from the local folder to the synchronized folder.

16. The system of claim **15**, wherein synchronizing the content item with the content management system comprises synchronizing the content item to one or more shared devices.

17. The system of claim **15**, wherein the operations further comprise identifying a location of the synchronized folder.

* * * * *